This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

Deep Learning for Spatio-Temporal Data Mining: A Survey

Senzhang Wang, Jiannong Cao, *Fellow, IEEE,* and Philip S. Yu, *Fellow, IEEE*

Abstract—With the fast development of various positioning techniques such as Global Position System (GPS), mobile devices and remote sensing, spatio-temporal data has become increasingly available nowadays. Mining valuable knowledge from spatio-temporal data is critically important to many real-world applications including human mobility understanding, smart transportation, urban planning, public safety, health care and environmental management. As the number, volume and resolution of spatio-temporal data increase rapidly, traditional data mining methods, especially statistics-based methods for dealing with such data are becoming overwhelmed. Recently deep learning models such as recurrent neural network (RNN) and convolutional neural network (CNN) have achieved remarkable success in many domains due to the powerful ability in automatic feature representation learning, and are also widely applied in various spatio-temporal data mining (STDM) tasks such as predictive learning, anomaly detection and classification. In this paper, we provide a comprehensive review of recent progress in applying deep learning techniques for STDM. We first categorize the spatio-temporal data into five different types, and then briefly introduce the deep learning models that are widely used in STDM. Next, we classify existing literature based on the types of spatio-temporal data, the data mining tasks, and the deep learning models, followed by the applications of deep learning for STDM in different domains including transportation, on-demand service, climate & weather analysis, human mobility, location-based social network, crime analysis, and neuroscience. Finally, we conclude the limitations of current research and point out future research directions.

Index Terms—Deep learning, Spatio-temporal data, Data mining.

1 Introduction

Spatio-temporal data mining (STDM) is becoming growingly important in the big data era with the increasing availability and importance of large spatio-temporal datasets such as maps, virtual globes, remote-sensing images, decennial census and GPS trajectories. STDM has broad applications in various domains including environment and climate (e.g. wind prediction and precipitation forecasting), public safety (e.g. crime prediction), intelligent transportation (e.g. traffic flow prediction), human mobility (e.g. human trajectory pattern mining), etc. Classical data mining techniques often perform poorly when applied to spatio-temporal datasets because of many reasons. First, ST data are usually embedded in continuous space, whereas classical datasets such as transactions and graphs are often discrete. Second, patterns of ST data usually present both spatial and temporal properties, which are more complex and the data correlations are hard to capture by traditional methods. Finally, one of the common assumptions in traditional statistics-based data mining methods is that data samples are independently generated. When it comes to the analysis of ST data, however, the assumption that the data samples are independent of each other usually does not hold because ST data tends to be highly self correlated.

Although STDM has been widely studied in the last several decades, a common issue is that traditional methods for STDM largely rely on feature engineering. In other words, conventional techniques for STDM are limited in their ability to process natural ST data in their raw form. For example, to analyze human's brain activity from fMRI data, usually careful feature engineering and considerable domain knowledge provided by the experts are needed to transform the raw data (e.g. the pixel values of the scanned fMRI images) into a suitable internal representation or feature vector. Recently deep leaning models such as recurrent neural network (RNN) and convolutional neural network (CNN) have achieved remarkable performance gains in various machine learning tasks due to the powerful ability in automatic feature representation learning. They have been broadly applied in many areas including computer vision, natural language processing, graph data mining, and STDM. Compared with traditional methods, the advantages of using deep learning models for STDM are as follows.

- **Automatic feature representation learning** Significantly different from traditional machine learning methods that require hand-crafted features, deep learning models can automatically learn hierarchical feature representations from the raw ST data. In STDM, the spatial proximity and the long-term temporal correlations of the data are usually complex and hard to capture. With the multi-layer convolution operation in CNN and the recurrent structure of RNN, such spatial proximity and temporal correlations in ST data can be automatically and effectively learned from the raw data.
- **Powerful function approximation ability** Theoretically, deep learning models can approximate any complex non-linear functions and can fit any curves

- *S.Z. Wang (szwang@nuaa.edu.cn) is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China, and also with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China.*
- *J.N. Cao is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China.*
- *P.S. Yu is with the Department of Computer Science, University of Illinois at Chicago, Chicago, USA.*

as long as they have enough layers and neurons. Deep learning models usually consist of multiple layers, and each layer can be considered as a simple but non-linear module with pooling, dropout, and activation functions so that it can transform the latent representation of features at a lower level into a more abstract feature representation at a higher level. With the composition of enough such transformations, very complex functions can be approximated to perform difficult STDM tasks with complex ST data.

- **Performing better with big data** Traditional machine learning methods like SVM and decision tree usually perform well on smaller datasets. However, the performance of such methods plateaus after hitting millions of training samples, and the margin of improvement in performance related to data size becomes minimal. In contrast, the performance of deep learning models can continue to increase when larger amounts of data are added. That means deep learning models can learn more and better use knowledge from the massive available ST data due to their powerful feature learning and function approximation abilities.

Figure 1 shows the number of yearly published papers that explore deep learning techniques for various STDM tasks. One can see that there is a significant increasing trend of the paper number in recent years. Only less than 10 related papers are published each year from 2012 to 2015. From 2016 on, the paper number increases rapidly and many researchers try different deep learning models for STDM in different applications domains. In 2018, there are about 90 related papers published, which is a large number. In view of the variety of applications and the richness of problems, there is an urgent need for overviewing the existing works that explore deep learning techniques in the rapidly advancing field of STDM due to the following reasons. It can highlight the similarities, differences and general frameworks of adopting deep learning models for addressing STDM problems. This enables the cross-pollination of methodologies and ideas across the research problems of different application domains. For example, it enables us to investigate whether and how the deep learning model designed for addressing a problem of a particular domain (e.g., traffic flow prediction in transportation) can be adopted to solve similar problems of other domains (e.g., crime prediction in crime analysis).

Related surveys on STDM As STDM has been studied for decades, several survey papers have reviewed relevant literature from different perspectives. [8] and [133] reviewed the STDM algorithms with a focus on discussing the computational issues in the application domains of remote sensing, climate science, and social media analysis. [78] focused on frequent pattern mining from ST data. It stated the challenges of pattern discovery from ST data and classified the patterns into three categories: individual periodic pattern, pairwise movement pattern and aggregative patterns over multiple trajectories. [18] reviewed the state-of-the-art in STDM research and applications, with emphasis placed on the data mining tasks of prediction, clustering and

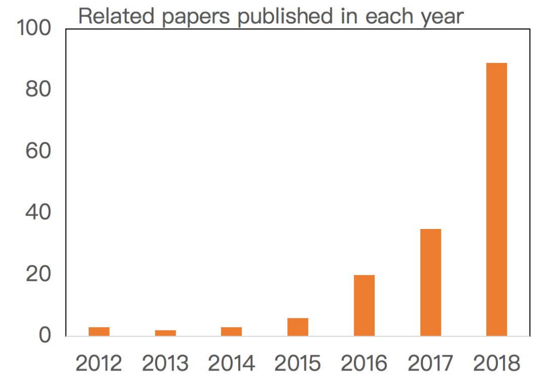


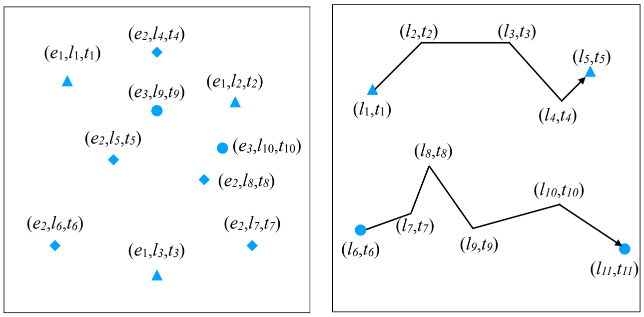
Fig. 1. Number of published papers that explore deep learning techniques for STDM in recent years.

visualization for ST data. [123] reviewed STDM from the computational perspective, and emphasized the statistical foundations of STDM. [104] reviewed the methods and applications for trajectory data mining, which is an important type of ST data. [66] provided a comprehensive survey on ST data clustering. [3] conducted a comprehensive review on the problems and methods in STDM. However, there are two major differences between this survey and the work [3]. First, [3] reviewed STDM from the perspective of traditional machine learning methods while our focus is on the application of deep learning methods on STDM. Thus the reviewed papers in the two surveys are quite different with small overlap. Second, the problems that are addressed by traditional methods and deep learning methods in the two surveys are different. Except for some general problems like predictive learning and anomaly detection, [3] focused on reviewing the problems of frequent pattern mining, clustering, change detection and relationship mining, which can be well addressed by traditional machine learning methods, while this survey focuses on the tasks of estimation & inference, classification, and other tasks. [106] and [150] provided surveys that specially focused on reviewing recent works on utilizing deep learning models for analyzing traffic data and performing various tasks for promoting intelligent transportation systems. There still lacks of a systematic survey on exploring deep learning techniques for STDM in general.

Our Contributions Compared with existing works, this paper makes notable contributions summarized as follows:

- **First survey** To our knowledge, this is the first survey that reviews recent works exploring deep learning techniques for STDM. In light of the rising number of related studies in recent years, we first categorize the ST data types, and then present the popular deep learning models that are widely used in STDM. We also summarize the data formats for different data types, and summarize which deep learning model is suitable to process what format of ST data.
- **Comprehensive survey** This survey provides a comprehensive overview on recent advances of using deep learning techniques for STDM tasks including predictive learning, classification, estimation and inference, anomaly detection, and others. For each task, we provide detailed descriptions on the rep-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering



(a) Three types of events

(b) The trajectories of two moving objects

Fig. 2. An illustration of event and trajectory data types

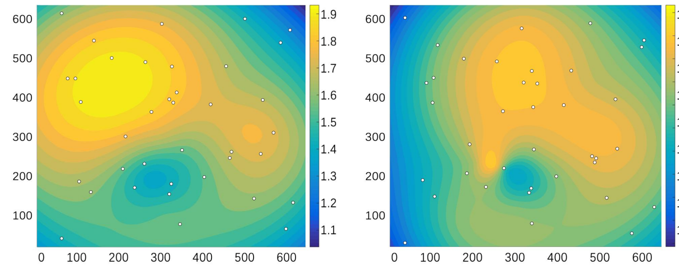


Fig. 3. An illustration of point reference data in two time stamps. The white circles are the locations of the sensors that record the readings. The color bars show the measurement distribution in the space.

resentative works and models for different types of ST data, and make necessary comparison and discussion. We also categorize and summarize current works according to the application domains including transportation, on-demand service, climate science, human mobility, location-based social network, crime & weather analysis, and neuroscience.

- **Future research directions** This survey also highlights several open problems that are not well studied currently, and points out possible research directions in the future.

Organization of This Survey The rest of this survey is organized as follows. Section 2 introduces the categorization of the ST data. Section 3 briefly introduces the deep learning models that are widely used for STDM, and summarizes which model is suitable for what type of ST data. Section 4 overviews various STDM tasks addressed by deep learning models. Section 5 presents a gallery of applications across various domains. Section 6 discusses the limitations of existing works and suggests future directions. We finally conclude this paper in Section 7.

2 Categorization of Spatio-Temporal Data

2.1 Spatio-Temporal Data Types

As the ways of data collection and representation for ST data differ significantly in real-world applications, we can categorize ST data into different types. Different application scenarios and ST data types lead to different categories of data mining tasks and problem formulations, and thus require different deep learning models. For example, CNN is designed to process image-like data, while RNN is usually used to process sequential data. Thus it is necessary to first categorize the general types of ST data and represent them properly. We follow and extend the categorization in [3], and classify the ST data into the following types: event, trajectory, point reference, raster, and video.

Event data. Event data such as crimes and traffic accidents are composed of discrete events that occur at certain locations in some time points. An event can be generally characterized by the type of the event, the location where the event occurred, and the occurrence time. For example, a traffic accident can be characterized as such a tuple (e, l, t), where e is type of the traffic accident, l is location of the accident and t is the time when it occurs. Fig. 2(a) shows an illustration of the event data. It shows three types of events denoted by different shapes of the symbol. Event data are common in many application domains such as criminology (e.g., urban crimes), epidemiology (e.g., disease outbreak events), transportation (e.g., traffic accidents), and social network (e.g., social event and trending topics).

Trajectory data. A trajectory is the moving path of an object in space over time (e.g., the travel path of a taxi trip). Trajectory data are usually obtained from the location sensors that are deployed on the moving objects, such as GPS on a taxi. The sensors record and transmit the locations of the object over time periodically. Fig. 2(b) shows an illustration of two trajectories. Each trajectory can be characterized as such a sequence $\{(l_1, t_1), (l_2, t_2)...(l_n, t_n)\}$, where l_i is the location (e.g., latitude and longitude) and t_i is the time when the moving object passes this location. With the development of mobile applications and IoT technologies, trajectory data such as human trajectories and urban traffic trajectories have become ubiquitous.

Point reference data. The measurements that are generated by a set of moving reference points in a particular space and time interval are referred to point reference data. For instance, meteorological data that are measured with weather balloons floating in space can be considered as point reference data. The weather balloons continuously sense and record the meteorological parameters such as temperature and humidity. Fig. 3 shows an example of the point reference data of sea surface temperature measured by the floating temperature sensors at two time stamps. As shown in the figure, the white circles are the locations of the temperature sensors at a certain time stamp. Note that the locations of the temperature sensors change over time.

Raster data. The measurements that are collected at fixed locations of a given space and at regular or irregular time points are referred to raster data. The major difference between raster data and point reference data is that the measurement locations of the point reference data keep changing while the locations of the raster data are fixed. The locations and times for the measurement can be regularly or irregularly distributed. Given m fixed locations $S = \{s_1, s_2, ...s_m\}$ and n time stamps $T = \{t_1, t_2, ...t_n\}$, a raster can be represented as a matrix $R^{m\times n}$, where each entry r_{ij} is the measurement at location s_i at time stamp t_j. Raster data are also common in many application domains such as transportation, climate science, and neuroscience.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

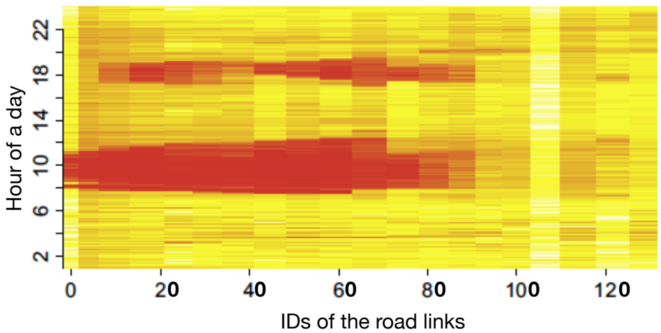


Fig. 4. An illustration of raster data collected from traffic flow sensors. The x-axis is the ID of the road links in a transportation network, and the y-axis is the hour of a day. Different colors denote different traffic flows on the road links captured by the road sensors.

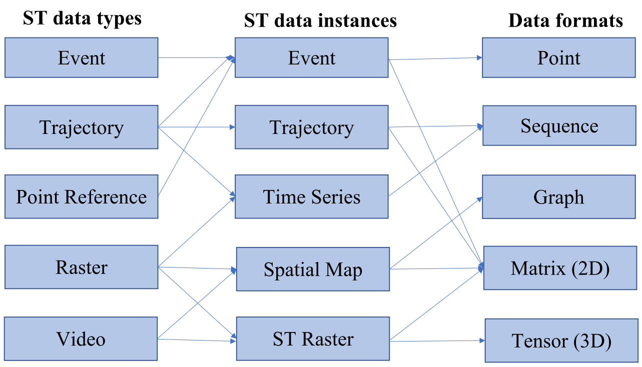


Fig. 5. Data instances and representations of different ST data types

For example, the air quality data (e.g., PM2.5) can be collected by the sensors deployed at fixed locations of a city, and the data collected in a continuous time period like one day form the air quality raster data. In neuroscience, functional magnetic resonance imaging (fMRI) is a popular technique to measure brain activity by detecting the changes of blood flowing past different active areas of the brain. The scanned fMRI signals form the raster data for analyzing the brain activity and identifying some diseases. Fig. 4 shows an example of the traffic flow raster of a transportation network. Each road is deployed a traffic sensor to collect real time traffic flows. The traffic volumes or speeds measured by all the road sensors in a whole day form a raster.

Video data. A video that consists of a sequence of images can be also considered as a type of ST data. In the spatial domain, the neighbor pixels usually have similar RGB values and thus present high spatial correlations. In the temporal domain, the images of consecutive frames usually change smoothly and present high temporal dependency. A video can be generally represented as a three dimensional tensor whose one dimension represents time t and the other two represent an image. Actually, video data can be also considered as a special raster data if we assume that there is a "sensor" deployed at each pixel and the RGB values continuously measured by all the sensors form the image frames. Video data analysis with deep learning techniques is extremely hot and a large number of papers are published. Although we categorize video as a type of ST data, we focus on reviewing related works from the perspective of data mining. Video data analysis falls into the research areas of computer vision and pattern recognition. Thus in this survey we do not cover the ST data type of video.

2.2 Data Instances and Formats

We call the basic data unit that a data mining or machine learning model operates upon as a data instance, such as a time series or an image. As we discussed before, the ST data can be categorized into different types, and thus can be formulated as different data instances. Even for the same type of ST data, different data instances can be extracted for different data mining tasks.

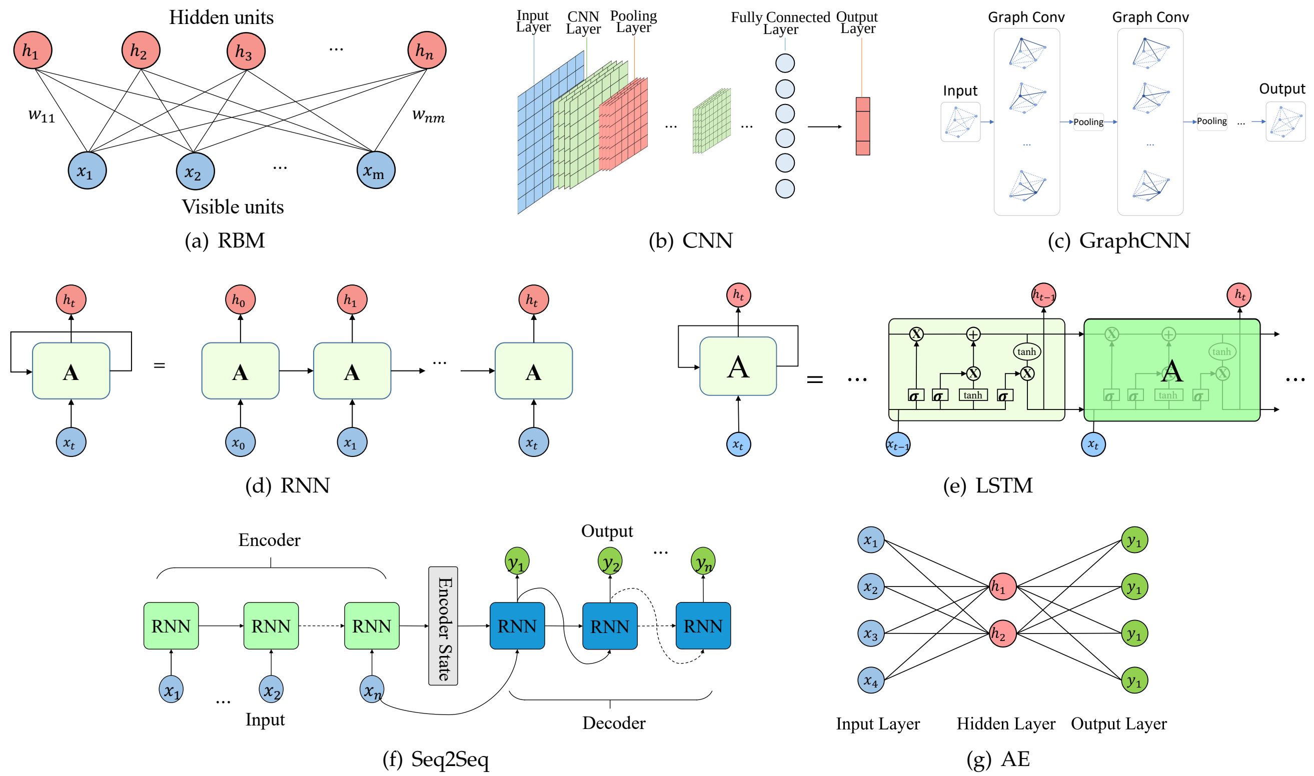
Data instances. In general, the data instances of ST data can be summarized into event, trajectory, time series, spatial maps, and ST raster as shown in the middle part of Fig. 5. Event is the simplest data instance, which usually can be represented as a tuple consisting of the location, time, and additional features such as event type. Besides event, trajectory and point reference can also be instantiated as event. For example, one can break a trajectory into several discrete points to count how many trajectories have passed a particular region in a particular time slot. Besides instantiated as point and trajectory, a trajectory can be also instantiated as time series in some applications. If we fix the location and count the number of trajectories traversing the location, a trajectory is instantiated as a time series. The data instance of spatial maps contains the measurements of the entire ST filed in a time period. For example, the traffic flows of all the loop sensors deployed at the expressway in time t is a spatial map. The data instance of ST raster contains the measurements covering the entire ST filed in all the time stamps.

Different data instances can be extracted from raster data such as time series, spatial maps or ST raster itself, depending on different applications and analytic requirements. First, one can consider the measurements of a single grid region in the space under study as a time series, and thus time series mining techniques can be applied for analyzing the data. Second, for each time stamp the measurement of a raster can be considered as a spatial map. Third, one can also consider all the measurements spanning all the locations and time stamps as a whole for analysis. In such a case, ST raster itself can be a data instance.

Data formats. For the above mentioned ST data instances, five types of data formats are generally utilized to represent them as the input of various deep learning models, which are point, sequence, graph, 2-dimensional matrix and 3-dimensional tensor as shown in the right part of Fig. 5. Different deep learning models require different data formats as input. Thus, how to represent the ST data instance depends on the data mining task and the selected deep learning model.

An event can be naturally represented as a point. A trajectory and a time series can be both represented as a sequence. Note that a trajectory sometime can be represented as a matrix whose two dimensions are the row and column IDs of the grid region. Each entry value of the matrix denotes whether the trajectory traverses the corresponding grid region. Such a data format is usually used to facilitate

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering



(a) RBM (b) CNN (c) GraphCNN

(d) RNN (e) LSTM

(f) Seq2Seq (g) AE

Fig. 6. Structures of the CNN and GraphCNN models

the utilization of CNN models [58], [110]. Although graph can be also represented as a matrix, here we categorize graph and image matrix as two different types of data formats. This is because graph nodes do not follow the Euclidean distance as the image does, and thus the ways of dealing with graphs and images are totally different. We will discuss more details on the methods of handling the two data formats later. Spatial maps can be both represented as graphs and matrices, depending on different applications. For example, in urban traffic flow prediction, the traffic data of a urban transportation network can be represented as a traffic flow graph [76], [147] or cell region-level traffic flow matrix [115], [130]. Raster data are usually represented as 2D matrices or 3D tensors. For the case of matrix, the two dimensions are the location and the time stamp, and for the case of tensor, the three dimensions are the row and column IDs of the cell region in the grid region map, and the time stamp. Matrix is a simpler data format compared with tensor, but it loses the spatial information. Matrix and tensor are both widely used to represent raster data. For example, in wind forecasting, the wind speed time series of multiple anemometers deployed in different locations are usually merged as a matrix, and then is feed into a CNN or RNN model for future wind speed prediction [86], [197]. In neuroscience, one's fMRI data are a sequence of scanned fMRI brain images, and thus can be represented as a tensor like a video. Many works use the fMRI images tensor as the input of a 3D-CNN model for feature learning to detect the brain activity [57], [67] and diagnose diseases [108], [151].

3 Deep Learning Models for Different Types of ST Data

3.1 Preliminary of Deep Learning Models

In this subsection, we will briefly introduce the deep learning models that are widely used for STDM, including RBM, CNN, GraphCNN, RNN, LSTM, AE/SAE, and Seq2Seq.

Restricted Boltzmann Machines (RBM). A RBM is a two-layer stochastic neural network [46] which can be used for dimensionality reduction, classification, feature learning and collaborative filtering. As shown in Fig. 6(a), the first layer of the RBM is called the visible layer with the neuron nodes $\{v_1, v_2, ...v_n\}$, and the second is the hidden layer with the neuron nodes $\{h_1, h_2, ...h_m\}$. As a fully-connected bipartite undirected graph, all nodes in RBM are connected to each other across layers by undirected weight edges $\{w_{11}, ...w_{nm}\}$, but no two nodes of the same layer are linked. The standard type of RBM has a binary-valued nodes and also bias weights. RBM tries to learn a binary code or representation of the input. Depending on the particular task, RBM can be trained in either supervised or unsupervised ways. RBM can be used for learning features for downstream STDM tasks.

Convolutional neural network (CNN). A typical CNN model usually contains the following layers as shown in Fig. 6(b): the input layer, the convolutional layer, the pooling layer, the fully-connected layer and the output layer. Some CNN structures also have the normalization layer after the pooling layer. The raw images are first input into the convolutional layer for high-level feature learning. The convolutional layer captures the high-level latent features through multiple filters called *kernel*. A *kernel* is usually a $k \times k$ square matrix, which moves in the input image matrix from left to right and from top to bottom. A filtering operation is performed on the corresponding positions of the input image matrix by the $kernel$ for generating high level features. The high level features are then input into the pooling layer, which performs downsampling operation along the spatial dimensionality to reduce the number of parameters. Finally, several fully-connected layers are stacked to perform non-linear transformation of the input latent features. Due to the powerful ability in capturing the correlations in the spatial domain, CNN is now widely used in learning from ST data, especially the data types of spatial maps and ST rasters.

GraphCNN. GraphCNN is recently widely studied to generalize CNN to graph structured data [153]. Fig. 6(c) shows a structure illustration of a GraphCNN model. The graph convolution operation applies the convolutional transformation to the neighbors of each node, followed by the pooling operation. By stacking multiple graph convolution layers, the latent embedding of each node can contain more information from neighbors which are multi-hops away. After the generation of the latent embedding of the nodes in the graph, one can either easily feed the latent embeddings to feed-forward networks to achieve node classification or regression goals, or aggregate all the node embeddings to represent the whole graph and then perform graph classification and regression. Due to its powerful ability in capturing the node correlations as well as the node features, GraphCNN can be used in mining graph structured ST data such as network-scale traffic flows and brain network data.

Recurrent neural network (RNN) and Long Short-Term Memory (LSTM) network. RNN is designed to recognize the sequential characteristics and use the previous patterns to predict the next likely scenario. They are widely used in the applications of speech recognition, natural language processing and time series data analysis. Fig. 6(d) shows the general structure of a RNN model, where X_t is the input data, A is the parameters of the network and h_t is the learned hidden state. One can see the output of the previous time step $t-1$ is input into the neural of the next time step t. Thus the historical information can be stored and passed to the future. A major issue of standard RNN is that it only has short-term memory due to the issue of vanishing gradients. To address this issue, LSTM network is proposed, which is able to learn long-term dependencies of the input data. As an extension of RNN, LSTM can remember the historical information of input over a much longer time period due to the specially designed memory unit as shown in the middle part of Fig. 6(e). As shown in the figure, a LSTM unit is composed of three gates: input gate, forget gate and output gate. The three gates control whether or not to let new input in (input gate), to ignore some unimportant information (forget gate) or to let it impact the current output (output gate). Both RNN and LSTM can deal with sequence and time series data for learning the temporal dependency.

Sequence to Sequence model (Seq2Seq). A Seq2Seq model aims to map a fixed length input with a fixed length output. Note that the length of the input and output can be different [131]. Although it is initially designed for the task of machine translation, Seq2Seq is a general framework and can be broadly applied to any sequence data related research problems such as speech recognition and online chatbot. As shown in Fig. 6(f), a Seq2Seq model generally consists of 3 parts: encoder, intermediate vector and decoder. Due to its powerful ability in capturing the dependencies among the sequence data, Seq2Seq model is widely used in ST prediction tasks where the ST data present high temporal correlations such as crowd flows and urban traffic data.

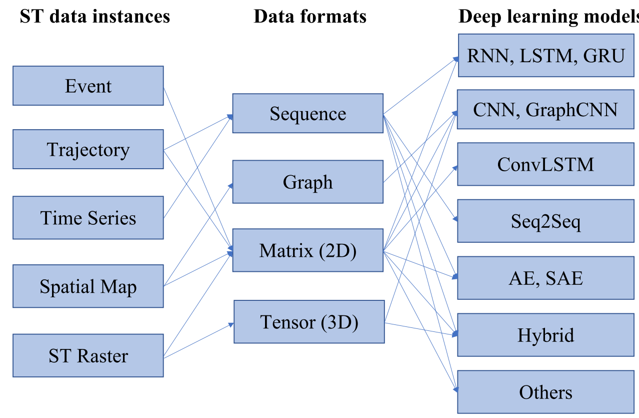


Fig. 7. Data formats for different deep learning models

Autoencoder (AE). An AE is a type of artificial neural network that aims to learn compact data codings for the input data in an unsupervised manner [46]. As shown in Fig. 6(g), AE first uses an encoder which contains one or multiple hidden layers to code the input as compact latent representation vectors. Then a decoder is used to reconstruct the input from the compact latent vector of the final hidden layer. AE learns a compact representation of the input data in the hidden layer or bottleneck layer, which can be considered as a way for dimensionality reduction. As an effective unsupervised feature representation learning technique, AE facilitates various downstream data mining and machine learning tasks such as classification and clustering. A stacked autoencoder (SAE) is a neural network consisting of multiple stacked AEs in which the outputs of the current AE is wired to the inputs of the successive AE [6].

3.2 Data Preparation and Deep Learning Model Selection

Given the raw ST data, the corresponding data instances are first constructed for data storage. As we discussed before, the ST data instances can be event, trajectory, time series, spatial maps, and ST raster as shown in Fig. 5. The constructed ST data instances are next formulated as proper data formats as the input of the deep learning model. The left part of Fig. 7 shows the ST data instances and their corresponding data formats. Trajectory and time series can be represented as sequence data. Trajectory can be also represented as a matrix so that CNN is applicable to learn the spatial features [21], [58], [94], [109]. Each entry of the matrix represents a cell region in the space under study, and the entry value denotes whether this cell region is covered by the trajectory. A spatial map is mostly represented as a 2D matrix, but sometimes it can also be represented as a graph. For example, the traffic sensors deployed in the express ways can be modeled as a graph where the nodes are the sensors and the edges are the road segments between two neighbor sensors. Then GraphCNN is applied to process the sensor graph and predict the future traffic (volume, speed, etc.) for all the nodes [76]. ST raster data can be represented as 2D matrices or 3D tensors. For example, the fMRI brain images of a person can be represented as a tensor and input into a 3D-CNN model for diseases classification [69], [108], while a correlation matrix can be constructed by extracting the time series correlations between pair-wise regions of the brain for brain activity analysis [42], [105].

The ST data of different formats are then feed into deep learning models for feature learning as shown in the right part of Fig. 7. Sequence data can be processed by the models of RNN, LSTM, Seq2Seq, AE and hybrid models. RNN, LSTM and Seq2Seq are all recurrent neural networks that are suitable to predict the sequence data. RNN and LSTM are usually used to predict the ST measurement in the next time slot. The difference is that LSTM can capture a much longer term dependencies of the input data compared with RNN. If a multi-step prediction is required, Seq2Seq model is more suitable. For example, in traffic prediction, a Seq2Seq model which consists of a set of LSTM units in the encoder layer and a set of LSTM units in the decoder layer can be applied to predict the traffic speed or volume in the next several time slots simultaneously [80].

As a feature learning model, AE or SAE can be used to various data formats to learn a low-dimensional feature coding. Sequence data can be first encoded as a low-dimensional feature with AE or SAE, and then further processed with other models. GraphCNN is designed to process the graph data for capturing the spatial correlations among the neighbor nodes. If the ST data are represented as graphs such as the traffic flows on a transportation network, GraphCNN can be applied [7], [76]. If the input ST data are represented as an image-like matrix that is Euclidean structured, CNN is a suitable choice. If the input is a sequence of image-like matrices, hybrid models that combine CNN and RNN such as ConvLSTM can be used [1], [61]. If the ST data is represented as a tensor or a sequence of tensors, 3D-CNN [11], [67] or the combination of 3D-CNN and RNN [108] can be used to learn the complex spatial and temporal dependencies of the data. Another important factor one needs to consider in deep learning model selection for STDM is the computational efficiency. The training of CNN and AE are much more efficient than RNN, Seq2Seq and GraphCNN. Therefore, the trade-off between efficiency and effectiveness needs to be carefully considered in deep learning model selection and design in a real application scenario.

Table 1 summarizes the works using deep learning models to process different types of ST data. As shown in the table, CNN, RNN and their variants (e.g. LSTM and GRU) are the most widely used deep learning models. CNN is mostly used to process the spatial maps and ST raster. Some works also used CNN to handle trajectory data. GraphCNN is specially designed to handle graph data. RNN models including LSTM and GRU can be broadly applied in dealing with trajectories, time series, and the sequences of spatial maps. ConvLSTM can be considered as a hybrid model which combines RNN and CNN, and is usually used to handle spatial maps. AE and SAE are mostly used to learn features from time series, trajectories and spatial maps. Seq2Seq is generally designed for sequential data, and thus only used to handle time series and trajectories. The hybrid models are also common for STDM. For example, CNN and RNN can be stacked to learn the spatial features first, and then capture the temporal correlations among the historical ST data. Hybrid models can be designed to fit all the four types of data formats. Other models such as network embedding [158], multi-layer perceptron [50], [182], generative adversarial nets (GAN) [83], [184], Residual Nets [69], [80] and deep reinforcement learning [43] have also been used for various STDM tasks recently.

4 Deep Learning Models for Addressing STDM Problems

In this section, we will categorize the STDM problems, and introduce the corresponding deep learning models to address them. Fig. 8 shows the distribution of various STDM problems addressed by deep learning models, including prediction, anomaly detection, classification, estimation & inference, recommendation and others. One can see the largest problem category is prediction, accounting for more than 70% works. This is because an accurate prediction largely relies on high quality features, while deep learning models are especially powerful in feature learning. Deep learning models are also used in other STDM tasks including classification, detection, estimation & inference, recommendation, etc. Next, we will introduce these STDM problems in detail and summarize the corresponding deep learning based solutions.

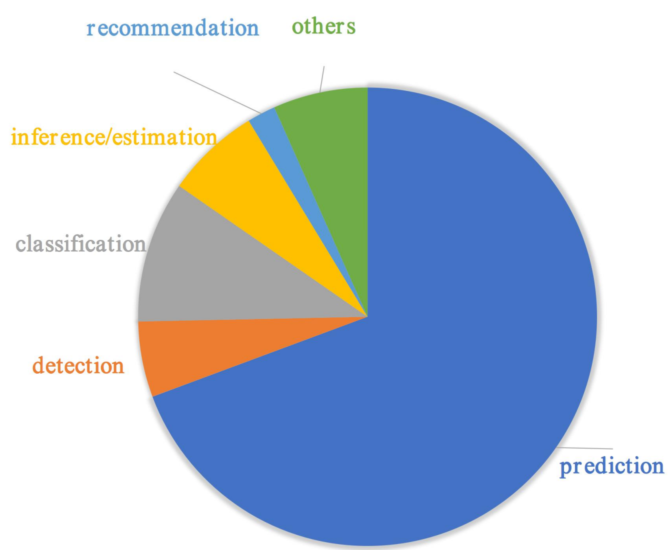


Fig. 8. Distribution of the STDM problems addressed by deep learning

4.1 Predictive Learning

The ST data instances differ for different ST data types as well as application domains, resulting in a variety of problem formulations in predictive learning. This subsection will review recent works that utilize various deep learning models for predictive learning. We will group these works based on the types of ST data instance that they deal with.

TABLE 1
Different deep learning models for processing four types of ST data.

	Trajectory	Time Series	Spatial Map	ST Raster
CNN	[21], [58], [94], [109]		[10], [14], [26], [47], [59], [60], [63], [71], [91], [105], [132], [138], [140], [145], [180], [196], [197]	[11], [65], [67], [69], [99], [108], [116], [121], [124], [139], [183]
GraphCNN			[7], [39], [76], [82], [84], [103], [134], [147], [149], [172], [184]	
RNN/LSTM	[29], [31], [32], [36], [37], [56], [68], [72], [79], [81], [90], [112], [128], [142], [152], [156], [157], [159], [160], [163], [177], [187], [188]	[13], [17], [28], [87], [92], [119], [173], [174], [175]	[2], [33], [118], [148], [155]	
ConvLSTM			[1], [61], [64], [89], [137], [143], [154], [195], [198]	
AE/SAE	[12], [107], [193]	[48], [95], [161]	[15], [42], [45], [178]	
RBM/DBN	[109]	[129]		[51], [57]
Seq2Seq	[73], [141], [164], [165]	[80]		
Hybrid	[100], [158]	[52], [86]	[5], [19], [75], [88], [101], [111], [127], [168], [169], [186]	[96], [120]
Others	[9], [23], [30], [40], [162], [171], [189], [191]	[16], [83], [117]	[20], [38], [70], [125], [126], [136], [179], [181], [182], [185], [199]	[55], [62]

Event. Events such as crimes [26], [50], [49], traffic accidents [198] and social events [38] are usually merged in temporal or spatial domains to form time series or spatial maps. [135] adapted ST-ResNet model to predict crime distribution over the Los Angeles area. Their model contains two staged. First, the raw crime point data are transformed as crime heat maps by merging all the crime events happened in the same time slot and the same region of the city. Then, the hierarchical structures of residual convolutional units are adopted to train a crime prediction model with the crime heat maps as input. Similarly, [50] proposed to use GRU model to predict the crime occurrences of a city. [198] proposed to use Convolutional Long Short-Term Memory model (ConvLSTM) for traffic accident prediction. The above two works first merge the point data of traffic accidents and model the traffic accident count in a spatio-temporal field as a 3D tensor. Each entry (i, j, t) of the tensor records the traffic accident count at the grid cell (i, j) in time slot t. The historical traffic accident tensors are input into CovnLSTM for prediction. [38] proposed a deep learning based multi-task predictive learning framework to effectively forecast the subtypes of future events happened at different locations.

Time series. In road-level traffic prediction, the traffic flow data on a road or freeway can be modeled as time series. Deep learning models can be used for road-level traffic flow prediction [95], [129]. [95] utilized stacked autoencoder to learn features from the traffic flow time series data for road-segment level traffic flow prediction. [129] modeled the traffic flow data at a freeway as time series and proposed to use Deep Belief Networks (DBNs) to forecast the future traffic flow based on the traffic flows of previous time slots. [119] studied the problem of taxi demand forecasting, and modeled the taxi demand at a particular area as a time series. A deep learning model with fully-connected layers was proposed to learn features from the historical time series of taxi demand, and then the features were integrated with other context features to predict the future demand. [16] proposed a novel Sequence-Aware Factorization Machine (SeqFM) for temporal predictive analytics. SeqFM utilizes a multi-layer network structure in order to exhaustively extract useful information from feature interactions in time series data.

RNN and LSTM are widely used for time series data prediction. The weather variables such as wind speed are usually modeled as time series and then RNN/LSTM models are applied for future weather forecasting [13], [17], [48], [87], [117], [175]. For example, [17] proposed an ensemble model to forecast wind speed. The model integrates traditional wind speed prediction models including wavelet threshold denoising (WTD) and adaptive neuro fuzzy inference system (ANFIS) with RNN. Note that the time series data usually do not contain the spatial information, and thus the spatial correlations among the data generated at different locations are not explicitly considered in such deep learning based prediction models.

Spatial map. The spatial maps can be usually represented as image-like matrices, and thus are suitable to be handled by CNN for various predictive learning tasks [60], [71], [180], [197]. [180] proposed a CNN based crowd flow forecasting model called UrbanFlow for real-time urban crow flow prediction. The input of the UrbanFlow model is the crowd flow spatial map. To forecast the supply-demand in ride-sourcing services, [60] proposed a hexagon-based convolutional neural network called H-CNN. Different from most previous works that normally partition a city area into a set of equal-sized grid cell regions, [60] partitions a city into various regular hexagon lattices. Wind speed data of one monitoring site can be modeled as time series, while the data of multiple sites can be represented as spatial maps. CNN models can be also applied to predict wind speed of multiple sites simultaneously [197]. [140] proposed Cross-Domain Convolutional Neural Network (CD-CNN) to recognize native/migrant attribute of an urban resident from mobile phone signaling data. People's locations are recorded in the mobile phone signaling data, and formulated as spatial maps as the input of CD-CNN.

To capture the temporal and spatial correlations of a sequence of spatial maps simultaneously, many works tried to combine CNN with RNN for the prediction. [154] proposed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

a convolutional LSTM (ConvLSTM) model for precipitation nowcasting, which combined the convolutional structure in CNN and the LSTM unites. ConvLSTM is a sequence-to-sequence prediction model, whose each layer is a ConvLSTM unit that has convolutional operations in both the input-to-state and state-to-state transitions. The input and output of the model are both spatial map matrices. Following this work, many works tried to apply ConvLSTM to other spatial map prediction tasks of different domains [1], [5], [24], [61], [64], [89], [143], [195]. [143] proposed a novel cross-city transfer learning method named *RegionTrans* for joint spatio-temporal data prediction among different cities. *RegionTrans* contains multiple ConvLSTM layers to catch the spatio-temporal patterns hidden in the data. [64] applied ConvLSTM network to predict precipitation by using multi-channel radar data. [195] proposed a deep neural network for predicting the pickup/dropoff demands of the passengers in mobility-on-demand (MOD) service in an end-to-end way. An encoder-decoder framework which is composed of convolutional and ConvLSTM units is employed to learn the features that capture both the spatio-temporal characteristics and pickup-dropoff interactions upon passenger demands. The passenger demands in the cell regions of a city are modeled as a spatial map and represented as a matrix. Similarly, [1] proposed a FCL-Net model which stacked ConvLSTM layers, standard LSTM layers and convolutional layers for passenger demand prediction in on-demand ride services. [89] proposed a unified neural network module Attentive Crowd Flow Machine (ACFM). ACFM is designed to predict the evolution patterns of the human mobility in urban areas through learning the dynamic representations of the ever changing urban crowd flow data. ACFM is composed of two ConvLSTM layers connected with a convolutional layer and an attention mechanism.

Other models can be also used for predicting spatial maps, such as GraphCNN [7], [82], [134], [184], [149], ResNet [136], [179], [181], Generative Adversarial Nets [185] and hybrid methods [101], [173], [169]. Note in this paper we consider that spatial maps contain both image-like data and graph data. Although graphs are also represented as matrices, they require totally different technique such as GraphCNN or GraphRNN. In road network-scale traffic prediction, the transportation network can be naturally modeled as a graph, and then GraphCNN or GraphRNN can be applied. [76] considered the traffic flow on a transportation network as the information propagating on a graph and invented Diffusion Convolutional Recurrent Neural Network (DCRNN) model for forecasting traffic conditions. Specifically, DCRNN adopts bidirectional random walks to capture the spatial dependency among the nodes on the graph, and uses an encoder-decoder architecture with scheduled sampling to capture the temporal dependency. [147] proposed a new topological framework called Linkage Network to model a road network. Based on the Linkage Network, a online Graph Recurrent Neural Network prediction model is developed to learn the propagation patterns of traffic flow on the road network. It simultaneously predicts traffic flow for all the road segments based on the information gathered from the whole graph. [134] introduced a spatio-temporal weighted graph (STWG) to first represent the sparse ST data. Then a scalable graph structured RNN is build on STWG to forecast the future ST data. [185] for the first time investigated how to combine CNN and adversarial learning for urban traffic flow prediction.

Trajectory. Two types of deep learning models, RNN and CNN are generally used for trajectory data prediction. A straightforward way to model a trajectory is to represent it as a sequence of locations as shown in Fig. 7. In such a case, RNN and LSTM models can be applied on the location sequences for prediction [32], [56], [68], [79], [112], [128], [156], [159]. [156] proposed Collision-Free LSTM for the position prediction of human trajectories. Collision-Free LSTM extended the classical LSTM by adding the repulsion pooling layer so that the hidden-states of neighboring pedestrians can be shared. [56] studied the human mobility prediction problem in the urban area. Given the previously visited locations of one person, an RNN based deep-sequence learning model was proposed to predict where he/her will go next. [128] proposed a DeepTransport model to predict the transportation mode such as walk, taking train, taking bus, etc., from a set of individual people's GPS trajectories. Four LSTM layers are used to construct DeepTransport for predicting a user's transportation mode in the future. [112] studied the bus travel time prediction problem. A RNN model with the LSTM block is introduced to learn the long-range dependencies in the bus trip trajectories for arrival time prediction.

A trajectory can be also represented as a matrix. In such a case, CNN can be applied to better capture the spatial correlations [58], [94]. [58] proposed a CNN-based approach for learning the representations of the semantic trajectories, which were then used to predict the possibly visited locations of the users in the future. In a semantic trajectory, each visited location is associated with a semantic meaning such as *home*, *work*, *shoppint*, etc. The semantic trajectories are modeled as a matrix whose two dimensions are semantic meanings and trajectory ID. Then the matrix is input into a CNN model to learn the latent features for next visited semantic location prediction. [94] modeled trajectories as two-dimensional images, where each pixel of the image represented whether the corresponding location was visited in the trajectory. Then multi-layer convolutional neural networks were adopted to combine multi-scale trajectory patterns for destination prediction of taxi trajectories. Modeling a trajectory as an image-like matrix is also utilized in other tasks such as anomaly detection and inference [103], which will be introduced in detail later.

ST raster. As we discussed before, ST raster can be represented as matrices whose two dimensions are location and time, or tensors whose three dimensions are cell region ID, cell region ID, and time. Usually for ST raster data prediction, 2D-CNN (matrices) and 3D-CNN (tensors) are applied, and sometimes they are also combined with RNN. [183] proposed a multi-channel 3D-cube successive convolution network named 3D-SCN to nowcast storm initiation, growth, and advection from the 3D radar data. [115] modeled the traffic speed data at multiple locations of a road in successive time slots as a ST raster matrix, and then input it into a deep neural network for traffic flows prediction. [99] explored the similar idea as [115] for traffic prediction on a large transportation network. [11] proposed a 3D Convolutional neural networks for predicting the vehicle flows of an entire city. Instead of predicting the traffic flow on a road, they try to predict vehicle flows in each grid region of a city. So they model the citywide vehicle flow data in successive time slots as ST rasters and input them into the proposed 3D-CNN model. Similarly, [124] proposed to model the mobility events of passengers in a city in different time slots as a 3D tensor, and then used the 3D-CNN model to predict the supply and demand of the passengers for transportation. Note that the major difference between ST raster and spatial map is that ST raster is the merged measurements in multiple time slots, while spatial map is the measurement in only one time slot. Thus the same type of ST data sometimes can be represented as both spatial maps and ST raster depending on the real application scenarios and the purposes of data analysis.

4.2 Classification

The classification task is mostly studied in analyzing fMRI data for disease identification. Recently, brain imaging technology is becoming increasingly important within the field of neuroscience, including functional Magnetic Resonance Imaging (fMRI), electroencephalography (EEG), and Magnetoencephalography (MEG) [114]. Particularly, deep learning models have become popular tools to analyze fMRI data for various classification tasks such as disease classification, brain function network classification and brain activation classification [151].

Various types of ST data can be extracted from the raw fMRI data depending on different classification tasks. [28] proposed to use LSTM model for classification of individuals with autism spectrum disorders (ASD) and typical controls with the resting-state fMRI time series data generated from different brain regions. [42], [45], [47], [62], [105], [125] modeled the fMRI data as spatial maps, and then used them as the input of the classification models. Instead of using each individual resting-state fMRI time-series data directly, [42] and [45] calculated the whole-brain functional connectivity matrix based on the Pearson correlation coefficients among the resting-state fMRI time-series data of different brain regions. Then the correlation coefficient matrix can be considered as a spatial map, and is input to a DNN model for ASD classification. [105] proposed a connectome-convolutional neural network (CCNN) for functional connectome classification. CCNN can be also conveniently adapted to other classification or regression tasks by changing the combinations of connectivity descriptors used to train the network model.

Some works also directly model the 3D structural MRI brain scanned images as the ST raster data, and then 3D-CNN model is applied to learn features from the ST raster for classification [55], [57], [69], [108], [121], [190]. [69] proposed two 3D convolutional network architectures for brain MRI classification, which were the modifications of a plain and residual convolutional neural networks. Their models can be applied to 3D MRI images without intermediate handcrafted feature extraction. [190] also used a 3D-CNN model for more accurate and efficient classification on the functional brain networks that were reconstructed from the 3D representation of the brain fMRI data.

4.3 Estimation and Inference

Current works on ST data estimation and inference mainly focus on the data types of spatial map and trajectory.

Spatial map. While monitoring stations have been established to collect pollutant statistics, the number of stations is usually very limited due to the high cost. Thus, inferring fine-grained urban air quality information is becoming an essential issue for both government and people. [19] studied the problem of air quality inference for any location based on the air pollutant of some monitoring stations. A deep neural network model ADAIN was designed to model the heterogeneous data and learn the complex feature interactions. In general, ADAIN combines two kinds of neural networks: i.e., feedforward neural networks to model static data and recurrent neural networks to model sequential data, followed by hidden layers to capture feature interactions. [132] investigated the application of deep neural networks on precipitation estimation based on the data of remotely sensed images. The stacked denoising autoencoder is employed to automatically extract latent features from the infrared cloud images and estimate the amount of precipitation. Given the origin and destination locations of a trip as well as the departure time, accurately estimating the trip duration is practically useful to many real applications. To address this issue, [74] proposed a deep multi-task representation learning model to estimate the arrival time of a trip. The underlying road network and the spatio-temporal prior knowledge are leveraged to generate more meaningful trip representations that preserve various trip properties.

Trajectory. [137], [177] studied the problem of travel time estimation from the mobility trajectory data. [177] proposed a RNN based deep model named DEEPTRAVEL which can learn from the historical trajectories to estimate the travel time. [137] proposed an end-to-end deep learning framework for Travel Time Estimation called DeepTTE that estimated the travel time of the whole trajectory directly rather than first estimating the travel times of individual road segments or sub-paths and then summing up them. [103] studied the problem of inferring the purpose of a user's visit at a certain location from trajectory data. They proposed a graph convolutional neural networks (GCNs) for the inference of activity types (i.e., trip purposes) from GPS trajectory data generated by personal smartphones. The mobility graphs of a user is constructed based on all his/her activity areas and connectivities based on the trajectory data, and then the spatio-temporal activity graphs are fed into GCNs for activity types inference. [37] studied the problem of Trajectory-User Linking (TUL), which aimed to link trajectories to the users who generated them. A RNN based model which combines the check-in trajectory embedding model and stacked LSTM is proposed to address the TUL problem. Identifying the distribution of users' transportation modes, e.g. *bike*, *train*, *walk* etc., is an essential part of travel demand analysis and transportation planning [21], [138]. [21] proposed a CNN model to infer travel modes based on only raw GPS trajectories, where the modes were labeled as *walk*, *bike*, *bus*, *driving*, and *train*. [141] proposed a novel deep hybrid trajectory recovery model which combines sequence-to-sequence generation learning and Kalman Filter to recover a high-sampled trajectory

TABLE 2
Related works in different application domains

Application domains	Related works
Transportation	[5], [14], [15], [21], [22], [27], [34], [41], [44], [53], [64], [70], [76], [80], [83], [93], [95], [96], [99], [100], [106], [109], [111], [112], [115], [118], [128], [129], [130], [138], [139], [147], [150], [167], [172], [173], [184], [185], [186], [196], [198]
On-demand Service	[1], [7], [39], [43], [60], [61], [71], [75], [77], [88], [119], [136], [168], [169], [195]
Climate & Weather	[10], [13], [17], [19], [40], [48], [63], [70], [84], [86], [87], [91], [116], [117], [120], [122], [126], [132], [154], [175], [182], [183], [197]
Human Mobility	[12], [29], [30], [31], [32], [36], [37], [56], [58], [73], [74], [85], [89], [90], [94], [103], [107], [110], [124], [128], [137], [141], [143], [144], [145], [152], [156], [157], [158], [163], [164], [165], [171], [176], [177], [179], [181], [187], [191], [193], [194], [199]
LBSNs	[9], [23], [68], [72], [90], [158], [159], [160], [162], [170], [188], [189]
Crime	[26], [50], [135]
Neuroscience	[28], [42], [45], [47], [51], [52], [55], [57], [59], [62], [67], [69], [102], [105], [108], [114], [121], [125], [151], [178]

based on the low-sampled trajectory in free space.

4.4 Anomaly Detection

Anomaly detection or outlier detection aims to identify the rare items, events or observations that differ remarkably from the majority of the data. Current works on anomaly detection for ST data mainly focus on the data types of event and spatial map.

Event. [130] tried to detect the non-recurring traffic congestions that are usually caused by traffic accidents, social events, extreme weathers, etc. A convolutional neural network was proposed to identify such non-recurring traffic anomalies. [186] studied how to use the deep learning method to detect traffic accidents with social media data. They first analyzed over 3 million tweets in Northern Virginia and New York City that were collected in one year, and then implemented two deep learning models DBN and LSTM to identify the traffic accident related tweets. [196] proposed to utilize Convolutional Neural Networks for automatically detecting the traffic incidents in urban transportation networks by using traffic flow data. [15] collected rich heterogeneous data including human mobility data and traffic accidents to study how human mobility affected the risk of traffic accident. A stack denoise Autoencoder model was proposed to learn the hierarchical feature representations of human mobility data, and the feature representations were used for predicting the risk level of traffic accident.

Spatial map. [91] presented how to apply deep learning techniques for climate extreme events detection such as hurricanes and heat waves. The model was trained to classify tropical cyclone, weather front and atmospheric river with the climate image data as the input. [63] studied the problem of extreme climate events detection and localization from the coarse climate data. The proposed method combines the following two deep neural network models, (1) a CNN model which is used for extreme climate events detection and localization, and (2) a pixel recursive super resolution model which is used for the reconstruction of the high resolution climate data from the low resolution climate data. To address the issue of limited labeled extreme climate events, [116] presented a multichannel spatiotemporal CNN model for extreme weather events identification. As a semi-supervised approach, the model in [116] leveraged the unlabeled data to address the data sparsity issue and improve the localization accuracy of the extreme climate events.

4.5 Other Tasks

Besides the tasks we discussed above, deep learning models are also applied in other STDM tasks including recommendation [9], [72], [142], [171], [189], pattern mining [110], relation mining [193], etc. [9] proposed a Content-Aware hierarchical POI Embedding model called CAPE for POI recommendation. From text contents, CAPE can capture not only the geographical influence of POIs, but also the characteristics of POIs. [189] also proposed to exploit the embedding learning technique to capture the contextual check-in information for POI recommendation. [142] employed an attention-based RNN and a value network for personalized route recommendation. [110] proposed a deep-structure model called DeepSpace to mine the human mobility patterns through analyzing the mobile data of human trajectories. [193] studied the problem of Trajectory-User Linking (TUL), which aimed to link trajectories to the corresponding users from the geo-tagged social media data. A semi-supervised trajectory-user relation learning framework TULVAE (TUL via Variational AutoEncoder) was proposed to learn the human trajectory representations in a neural generative architecture. [194] proposed a Multi-Context Trajectory Embedding Model (MC-TEM) to learn the distributed representations of trajectories through exploring contextual information. MC-TEM is applied to address the tasks of location recommendation and social link prediction.

5 Applications

In this section, we will categorize and summarize related works in different application domains including transportation, on-demand service, climate & weather, human mobility, location-based social networks (LBSNs), crime analysis, and neuroscience. Table 2 groups the recent works into seven types of applications mentioned above. Next, we will introduce these applications in detail.

5.1 Transportation

With the rising availability of transportation data collected from various sensors like loop detector, road camera, and

GPS probes, there is an urgent need to utilize deep learning methods to learn the complex and highly non-linear spatio-temporal correlations among the traffic data to facilitate various tasks such as traffic flow prediction [53], [115], [129], [161], [111], [185], traffic incident detection [118], [186], [196] and traffic congestion prediction [100], [130].

Transportation data usually contain information of the traffic speed, volume, or traffic incidents, the road segments or regions, and time. Transportation data can be the ST data types of event, time series, spatial maps and ST raster in different application scenarios. For example, in road network-scale traffic flow prediction the traffic flow data collected from multiple road loop sensors can be modeled as a raster matrix where one dimension is the locations of the sensors and the other is the time slots [99]. The loop sensors can be also connected as a sensor graph based on the connections among the road links where the sensors are deployed. Thus the traffic data of the road sensor network can be modeled as a graph so that GraphCNN models can be applied [76], [172]. While in road-level traffic prediction, the historical traffic flow data on each single road is modeled as a time series, and then RNN or LSTM are used for traffic prediction on a single road [53], [161]. As the urban traffic data are highly correlated to many external context factors such as weather, holiday, social events, and hour of a day, the design of deep learning models should carefully consider how to fuse such external features with the transportation related ST data as we discussed before [15], [80].

5.2 On-Demand Service

In recent years, various on-demand service platforms such as Uber, Mobike, DiDi, GoGoVan have become increasingly popular due to the wide use of mobile phones. The on-demand services have taken over the traditional businesses by serving people in real time with what and where they want. Many on-demand services produce a large volume of ST data which involve the locations of the customers and the required service time. For example, Uber and DiDi are two popular ride-sharing on-demand service providers in USA and China, respectively. They provide services including taxi hailing, private car hailing, and ride-sharing to users via smartphone applications.

To better meet customers' demand and improve the service, a crucial problem is to accurately predict the demand and supply on the service at different locations and time [1], [7], [119], [168], [136], [71], [60], [195], [61], [39], [43], [77], [88]. [1] proposed to apply deep learning methods to forecast the demand-supply distributions of the dockless bike-sharing system. [82] proposed a graph CNN model to predict the hourly demand on bikes of each bike station in a bike-sharing system. As the bike trips among all the bike stations form a bike flow graph, a graph CNN model is used in [82]. [119], [168] proposed to use LSTM model to predict the taxi demand in different areas. [136] applied ResNet model to predict the supply-demand on the service of online car-hailing. In the application of on-demand service, usually the demand-supply on the service in different regions of the city are modeled as spatial maps or raster tensors. Then they are input into CNN, RNN and hybrid models for feature learning and future supply-demand prediction.

5.3 Climate & Weather

The weather data usually contain the atmospheric and oceanic conditions (e.g., temperature, wind speed, pressure, precipitation, and air quality) that are measured by various climate sensors deployed at fixed or floating locations. As the climate data of different locations usually present high spatio-temporal correlations, STDM techniques are widely used for short-term and long-term weather forecasting. Especially, with the recent advances of deep learning techniques, many works tried to design deep learning models for analyzing various climate and weather data [70], [122], such as air quality inference [19], [84], precipitation prediction [91], [154], wind speed prediction [86], [197], and extreme weather detection [91]. The data types related to climate and weather can be spatial maps (e.g. radar reflectivity images) [183], time series (e.g. wind speed of an anemometer) [17], and events (e.g. extreme weather events) [91]. Most works in this application area focus on the tasks of climate or weather forecasting [19], [40], [84], [86], [197], [122] and extreme weather detection [91]. For example, [19] developed a neural attention model to predict the urban air quality based on the observations of multiple air quality monitoring stations. [91] proposed to use CNN model for detecting extreme weather in climate databases. CNN model can be also used to predict the precipitation from the remote sensing images [132].

5.4 Human Mobility

The large volume of human mobility data combined with data mining models enable us to quantitatively study human mobility patterns, and help us have a better understanding on the spatio-temporal structures and regularities of human mobilities. Mining the human mobility data is practically important for applications including traffic forecasting, urban planning, and human behavior analysis. Deep learning techniques applied on human mobility analysis mostly focus on the problems of trajectory data mining and urban crowd flow data prediction.

For trajectory data mining, the studied tasks include trajectory representation learning [194], [73], [164], [158], [37], [144], [152], [12], trajectory classification [30], trajectory prediction [58], [32], [56], [156], [29], [90], [163], [94], trajectory pattern mining [110], [107], human transportation mode inference from trajectories [21], [37], [128], and travel time estimation [177], [74], [137], [157]. Based on different application scenarios and analytic purposes, human trajectories can be modeled as different ST data types and formats so that suitable deep learning models can be applied. The most widely used models for human trajectory data mining are RNN an CNN models, and sometimes the two types of models are combined as hybrid methods to capture both the spatial and temporal correlations of the human mobility data. Different from trajectory data that records a sequence of locations and time in each trip, crowd flow data only have the start and end locations of a trip, and how many people flow in and out a particular region can be counted. There are many recent works using deep learning methods for urban crowd flow prediction and crowd density estimation [181], [199], [85], [179], [143], [89].

5.5 Location-Based Social Networks (LBSNs)

LBSNs such as Foursquare and Flickr are social networks that use GPS features to locate the users and let the users broadcast their locations and other contents from their mobile device [192]. LBSNs do not merely mean that the locations are added to the user generated contents (UGC) in social networks so that people can share their location information, but also reshape the social structure among individual users that are connected by both their locations in the physical world and their location-tagged social media content in the virtual world. LBSNs contain a large number of user check-in data which consists of the instant locations of each user. [170] for the first time introduced the spatio-temporal dimension into the community detection in LBSNs. Currently, various deep learning models are designed to analyze the user generated ST data in LBSNs, and the studied tasks include next check-in location prediction or recommendation [58], [9], [189], [68], [72], [90], [188], user representation learning in LBSNs [158], geographical feature extraction [23] and user check-in time prediction [159], [160]. Different from the ST data of other applications, the ST data generated in LBSNs have rich social relation information. Thus when performing STDM tasks in LBSNs, the social relations are usually helpful and needed to be considered in the designed deep learning models.

5.6 Crime Analysis

More and more cities make the crime data held by law enforcement agencies publicly available for research purposes. The crime data typically contain the crime type, time and location of the crime, victims, criminals or suspect, and some other information. As a typical ST event data, the crimes occurred at different locations of a city and in different time slots of a day usually present high spatio-temporal correlations. Deep learning models are used with the crime number heat map of a city as the input to capture such complex correlations for crime prediction [26], [50], [135]. For example, [26] proposed a Spatiotemporal Crime Network based on CNN to forecast the crime risk of each region in the urban area for the next day. [135] proposed to utilize ST-ResNet model to collectively predict crime distribution over the Los Angeles area. [50] developed a deep neural network model named DeepCrime for urban crime prediction. DeepCrime can predict the crime occurrences of different categories by considering the dependencies among the crime categories. As we discussed before, although crime data belong to the ST data type of event, but are usually represented as spatial maps through merging the data in spatial or temporal domains so that deep learning models can be applied for analytics.

5.7 Neuroscience

Various brain imaging technologies such as functional Magnetic Resonance Imaging (fMRI), electroencephalography (EEG), Magnetoencephalography (MEG), and functional Near Infrared Spectroscopy (fNIRS) are widely applied within the field of neuroscience. The spatial and temporal resolutions of the brain activity measured by these technologies differ significantly. fMRI measures the neural activity from millions of locations, while EEG only measures from tens of locations. fMRI typically performs one measurement for every two seconds, while the temporal resolution of EEG is around 1 millisecond.

The fMRI and EEG imaging data can be naturally represented as spatial maps or rasters, and thus are suitable to be handled by deep learning models for addressing many research issues in neuroscience. [102], [114], [151] have reviewed recent works that apply deep learning techniques in processing neuroimaging data and related applications in neuroscience. As we discussed before, deep learning models are mostly used for the classification task in neuroscience such as disease classification or diagnosis [28], [42], [45], [57], [69], [121], [125], brain function network based classification [59], [62], [105] and brain activation classification [55], [178]. Various deep learning models can be used for analyzing the neuroimaging data in these tasks. For example, LSTM was used to identify Autism Spectrum Disorder (ASD) [28], CNN was used to diagnose amnestic Mild Cognitive Impairment (aMCI) [105] and FNN was used to classify Schizophrenia [113].

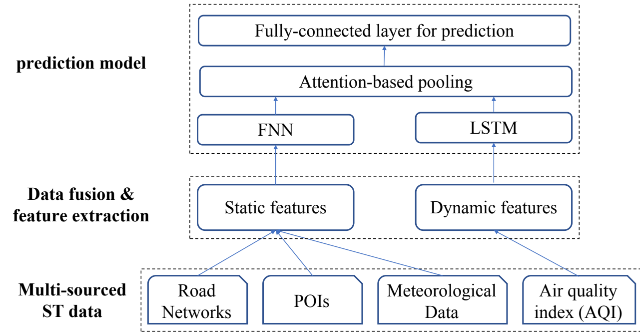


Fig. 9. ADAIN fuses ST data and urban data in the data level.

6 Discussions

6.1 Fusing Multi-Sourced Spatio-Temporal Data

Besides the ST data under study, there are usually some other types of data that are highly correlated to the ST data. Fusing such data together with the ST data can usually improve the performance of various STDM tasks. For example, the urban traffic flow data can be significantly influenced by some external contexts such as weather, social events, and holidays. Some recent works try to fuse the ST data and other types of data into a deep learning architecture for jointly learning features and capturing the correlations among them [15], [19], [80], [168], [174], [183], [198]. Generally, there are two popular ways to fuse the multi-sourced data in applying deep learning models for STDM, raw data-level fusion and latent feature-level fusion.

Raw data-level fusion. For the raw data-level fusion, the multi-sourced data are integrated first and then input into the deep learning model for feature learning. [198] proposed to apply the Convolutional LSTM (ConvLSTM) for traffic accident prediction. First, the area under the study like a city is divided into equal-sized grid cell regions. Then a variety of features including the traffic volume in each cell region,

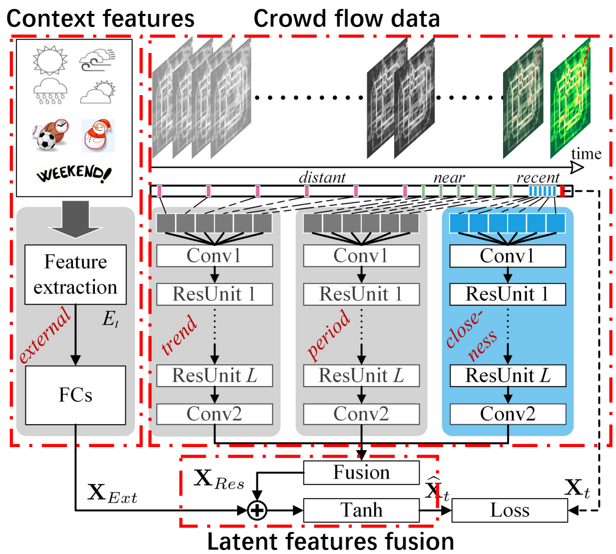


Fig. 10. ST-ResNet fuses ST data and context features in the latent feature level [80].

road condition, weather, and satellite images are extracted and mapped to the corresponding cell regions. Given the number of traffic accidents and the extracted features of each cell region as the model input, a Hetero-ConvLSTM model is proposed to predict how many accidents will occur in each cell region in the future. [19] proposed the ADAIN model which fused both the urban air quality information from monitoring stations and the urban data including POIs, road networks and meteorology for inferring fine-grained urban air quality of a city. As shown in Fig 9, the air quality index (AQI) data and the urban data including road networks, POIs, meteorological data are fused through concatenating their features. Then all the features are fed into FNN and RNN models for prediction.

Feature-level fusion. For the latent feature-level fusion, different types of raw data are first input into different deep learning models to learn latent feature representations separately, and then a feature fusion component is designed to fuse all the latent features together. [80] extended the residual neural network and proposed ST-ResNet to collectively forecast the inflow and outflow of crowds in different regions of a city. As shown in Fig 10, ST-ResNet fuses the latent features from two types of data, the crowd flows in a city and the external contexts including the weather and holiday events. Two components are designed to learn the latent features of the external features and the crowd flow data features separately, and then a feature fusion function $tanh$ is used to integrate the two types of learned latent features. [168] designed a Deep Multi-View Spatio-Temporal Network (DMVST-Net) which combined multi-view data for taxi demand prediction. DMVST-Net integrates the data from three different views: temporal view, spatial view and semantic view. CNN and LSTM are used to learn the latent features from the spatial and temporal views, respectively. Network embedding is applied to learn the semantic correlations among different regions. Finally, the latent features of the three views are fused together by a fully connected layer and are used for taxi demand prediction.

6.2 Attention Mechanism

Attention mechanism is initially designed to improve the performance of the Encoder-Decoder RNN on the task of machine translation [4]. Encoder-Decoder RNN generally encodes the input sequence to a fixed length internal representation vector. As the length of the input sentence varies significantly, Encoder-Decoder RNN may not learn a good representation vector if the input sequence is very long. To address this issue, attention mechanism is developed to enable the model to learn which encoded words in the input sequence to pay attention to and to what degree during the prediction of each word in the output sequence. Although attention is initially proposed in machine translation with the word sequence data as the input, it actually can be applied to any kind of inputs such as images, which is called visual attention. As many ST data can be represented as sequential data (e.g., time series and trajectories) and image-like spatial maps, attention can also be incorporated into deep models for improving the performance of various STDM tasks [19], [32], [33], [50], [72], [79], [89], [195].

The neural attention mechanism used in STDM can be generally categorized into spatial domain attention [19], [33], temporal domain attention [32], [50], [72], [195], and spatial-temporal domain attention [79], [89]. [33] proposed a combined attention model in the spatial domain. It utilizes both "soft attention" and "hard-wired attention" to learn the trajectory representations in local neighborhood areas for predicting the possible future positions of the pedestrian. [50] proposed an attentive hierarchical recurrent network model named DeepCrime for crime prediction. The temporal domain attention mechanism is applied in [50] to capture the crime patterns from historical data for facilitating the prediction of crime occurrences in the future. In the proposed attention mechanism, the importance weights of the crimes occurred in previous time slots are estimated through a softmax function. The weights are then automatically assigned to the corresponding hidden states of the model at different previous time slots. [79] proposed a multi-level attention network for predicting the geo-sensory time series data. These data are usually generated by the sensors that are deployed in different geospatial locations for cooperatively monitoring the environmental parameters in real time, such as Air Quality Index (AQI). Specifically, in the first level attention, a spatial attention mechanism consisting of local spatial attention and global spatial attention is proposed to capture the complex spatial correlations between the time series data of different sensors. In the second level attention, a temporal attention is applied to capture the dynamic temporal correlations among different time intervals in a time series.

6.3 Deep Learning Models *vs* Traditional Models

We discuss the comparison between deep learning models and traditional models in STDM. As shown in Table 3, we compare them from the perspectives of ST features, ST data types, STDM tasks, temporal dependency, spatial dependency, interpretability and domain knowledge.

Deep learning models can perform automatic ST feature learning from raw data in both supervised [46] and unsupervised manners [131]. For traditional machine learning

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

TABLE 3
Deep learning models *vs* Traditional models in STDM

	Deep learning models	Traditional models
ST features	Automatic learning	Hand-crafted
ST data types	n-to-n	1-to-1
STDM tasks	Prediction, classification, estimation, etc.	Prediction, classification, estimation, clustering, frequent pattern mining, change detection, etc.
Temporal dependency	Long/Short term	Short term
Spatial dependency	Global/Local	Local
Interpretability	Low	High
Domain knowledge	Little	Much

models, ST features usually need to be manually extracted. As summarized in Table 1, one type of deep learning model can process multiple types of ST data. For one type of ST data, multiple types of deep learning models can be selected for feature learning. For example, trajectory can be handled by CNN [21], RNN [29], LSTM [36], AE [12], Seq2Seq [73], and hybrid models [158], while CNN can be used to handle the ST data types of trajectory [58], spatial map [10], and ST raster [11]. Due to limited feature learning capacity, one shallow model can usually process one type of ST data. For the STDM tasks, deep learning models are mostly applied in the tasks of prediction, classification and estimation as shown in Fig. 8. Traditional models have broader application domains in STDM. Besides the tasks addressed by DL models, the tasks of clustering, frequent pattern mining, change detection, and relationship mining are also widely studied by previous works with traditional data mining methods [3].

From the perspective of learning temporal dependencies, deep learning models can capture both long term and short term temporal dependencies. For example, ST-ResNet can model the temporal closeness, period, and long-term trend of human mobility data simultaneously for citywide crowd flow prediction [179]. The short term (temporal closeness and period) and long term (trend) temporal features are fused through using stacked convolution layers and residual units. It is very difficult for traditional machine learning methods such as regression models to capture the multi-scale and highly nonlinear temporal dependencies. For the spatial domain, deep learning models are also more effective in capturing local and global spatial correlations of the ST data. For example, in traffic prediction the transportation network is usually modeled as a graph. The local and global spatial correlations among the road links can be captured by multi-layer convolutional aggregators or attention aggregators of GraphCNN for traffic prediction [76]. Traditional traffic prediction models such as coupled hidden Markov model can only capture the local spatial correlation among the neighbor road links [146]. In terms of interpretability, deep learning models are usually considered as *black boxes* whose design does not rely on much domain knowledge, and thus their interpretability is lower than traditional machine learning models. To address this issue, some recent works tried to inject more domain knowledge into deep learning models to improve their interpretability. For example, in the analysis of medical time series data, researchers have tried to embed the medical knowledge into the designed neural network structures [35], add specific penalty items to the loss functions of the deep learning models [97], and set appropriate weights for input or hidden layer features [98].

7 Open Problems

Though deep learning models have been broadly used for STDM in diverse application domains discussed above, challenges still exist due to the highly complex, large volume, and fast increasing ST data. In this section, we provide some open problems that have not been well addressed by current works and need further studies in the future.

Interpretable models. Current deep learning models for STDM are mostly considered as *black-boxes* which lack of interpretability. Interpretability gives deep learning models the ability to explain or to present the model behaviors in understandable terms to humans, and it is an indispensable part for machine learning models in order to better serve people and bring benefits to society [25]. Considering the complex data types and representations of ST data, it is more challenging to design interpretable deep learning models compared with other types of data such as images and word tokens. Although attention mechanisms are used in some previous works to increase the model interpretability such as periodicity and local spatial dependency [19], [50], [79], how to build a more interpretable deep learning model for STDM is still not well studied and remains an open problem.

Deep learning model selection. For a given STDM task, sometimes multiple types of related ST data can be collected and different data formats can be chosen. How to properly select the ST data formats and the corresponding deep learning models is not well studied. For example, in traffic flow prediction, some works model the traffic flow data of a road as a time series so that RNN, DNN or SAE are used for prediction [95], [129]; some works model the traffic flow data of multiple road links as spatial maps so that CNN is applied for prediction [180]; and some works model the traffic flow data of a road network as a graph so that GraphCNN is adopted [76]. There is a lack of deeper studies on how to properly select deep learning models and data formats of the ST data for better addressing the STDM task under study.

Broader applications to more STDM tasks. Although deep learning models have been successfully applied to address many types of STDM tasks discussed above, there are still some tasks that have not been fully addressed by deep learning models such as frequent pattern mining and relationship mining [3], [78]. The major advantage of deep learning is its powerful feature learning ability, which is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

essential to some STDM tasks such as predictive learning and classification that largely rely on high quality features. However, for some STDM tasks like frequent pattern mining and relationship mining, learning high quality features may not be that helpful. Based on our review, currently there are very few or even no works that utilize deep learning models for the two tasks mentioned above. So it remains an open problem that how deep learning models alone or the integration of deep learning models with traditional models such as frequent pattern mining and graphical models can be extended to broader applications and more STDM tasks.

Fusing multi-modal ST datasets. In big data era, multi-modal ST datasets are increasingly available in many domains such as neuroimaging, climate science and urban transportation. For example, in neuroimaging, fMRI and DTI can both capture the imaging data of the brain activity with different technologies that provide different spatio-temporal resolutions [54]. How to use deep learning models to effectively fuse them together to better perform the tasks of disease classification and brain activity recognition is less studied. The multi-modal transportation data including taxi trajectory data, bike-sharing trip data and public transport check-in/out data of a city can all reflect the mobility of urban crowd flow from different perspectives. Fusing and analyzing them together rather than separately can more comprehensively capture the underlying mobility patterns and make more accurate predictions. Although there are recent attempts that try to apply deep learning models for transferring knowledge from the crowd flow data among different cities [143], [166], how to fuse multi-modal ST datasets with deep learning models is still not well studied and needs more research attention in the future.

8 Conclusion

In this paper, we conduct a comprehensive overview of recent advances in exploring deep learning techniques for STDM. We first categorize the different data types and representations of ST data and briefly introduce the popular deep learning models used for STDM. For different types of ST data and their representations, we show the corresponding deep learning models that are suitable to handle them. Then we overview current works based on the categorization of the ST data and the STDM tasks including predictive learning, representation learning, classification, estimation and inference, anomaly detection, and others. Next we summarize the applications of deep learning techniques for STDM in different domains including transportation, on-demand service, climate & weather, human mobility, location-based social networks (LBSNs), crime analysis, and neuroscience. Finally, we list some open problems and point out the future research directions for this fast growing research filed.

Acknowledgment

This work is supported by National Key R&D Program of China (No.: 2018YFB1003900), HK RGC Collaborative Research Fund (No.: C5026-18G), HK RGC Research Impact Fund (No.: R5034-18), CCF-Tencent Open Research Fund, the Fundamental Research Funds for the Central Universities (No.: NZ2020014) and NSF grants (Nos.: III-1763325, III-1909323, and SaTC-1930941).

References

[1] Y. Ai, Z. Li, M. Gan, Y. Zhang, D. Yu, W. Chen, and Y. Ju. A deep learning approach on short-term spatiotemporal distribution forecasting of dockless bike-sharing system. *Neural Computing and Applications*, pages 1–13, 2018.

[2] A. Akbari Asanjan, T. Yang, K. Hsu, S. Sorooshian, J. Lin, and Q. Peng. Short-term precipitation forecast based on the persiann system and lstm recurrent neural networks. *Journal of Geophysical Research: Atmospheres*, 123(22):12–543, 2018.

[3] G. Atluri, A. Karpatne, and V. Kumar. Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys (CSUR)*, 51(4):83, 2018.

[4] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[5] J. Bao, P. Liu, and S. V. Ukkusuri. A spatiotemporal deep learning approach for citywide short-term crash risk prediction with multi-source data. *Accident Analysis & Prevention*, 122:239–254, 2019.

[6] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, 2006.

[7] D. Chai, L. Wang, and Q. Yang. Bike flow prediction with multi-graph convolutional networks. In *SIGSPATIAL*, 2018.

[8] V. Chandola, R. R. Vatsavai, D. Kumar, and A. Ganguly. Analyzing big spatial and big spatiotemporal data: a case study of methods and applications. *Big Data Analytics*, 33(239), 2015.

[9] B. Chang, Y. Park, D. Park, S. Kim, and J. Kang. Content-aware hierarchical point-of-interest embedding model for successive poi recommendation. In *IJCAI*, 2018.

[10] A. Chattopadhyay, P. Hassanzadeh, and S. Pasha. A test case for application of convolutional neural networks to spatio-temporal climate data: Re-identifying clustered weather patterns. *arXiv preprint arXiv:1811.04817*, 2018.

[11] C. Chen, K. Li, S. G. Teo, G. Chen, X. Zou, X. Yang, R. C. Vijay, J. Feng, and Z. Zeng. Exploiting spatio-temporal correlations with multiple 3d convolutional neural networks for citywide vehicle flow prediction. In *ICDM*, 2018.

[12] C. Chen, C. Liao, X. Xie, Y. Wang, and J. Zhao. Trip2vec: a deep embedding approach for clustering and profiling taxi trip purposes. *Personal and Ubiquitous Computing*, pages 1–14, 2018.

[13] M. Chen, J. M. Davis, C. Liu, Z. Sun, M. M. Zempila, and W. Gao. Using deep recurrent neural network for direct beam solar irradiance cloud screening. In *Remote Sensing and Modeling of Ecosystems for Sustainability XIV*, volume 10405, 2017.

[14] M. Chen, X. Yu, and Y. Liu. Pcnn: Deep convolutional networks for short-term traffic congestion prediction. *IEEE Transactions on Intelligent Transportation Systems*, (99):1–10, 2018.

[15] Q. Chen, X. Song, H. Yamada, and R. Shibasaki. Learning deep representation from big and heterogeneous data for traffic accident inference. In *AAAI*, pages 338–344, 2016.

[16] T. Chen, H. Yin, Q. V. H. Nguyen, W.-C. Peng, X. Li, and X. Zhou. Sequence-aware factorization machines for temporal predictive analytics. In *Proceedings of ICDE*, 2020.

[17] L. Cheng, H. Zang, T. Ding, R. Sun, M. Wang, Z. Wei, and G. Sun. Ensemble recurrent neural network based probabilistic wind speed forecasting approach. *Energies*, 11(8):1958, 2018.

[18] T. Cheng, J. H. B. Anbaroglu, and G. Tanakaranond. Spatiotemporal data mining. *Handbookd of Regional Science*, pages 1173–1193, 2014.

[19] W. Cheng, Y. Shen, Y. Zhu, and L. Huang. A neural attention model for urban air quality inference: Learning the weights of monitoring stations. In *AAAI*, 2018.

[20] O. Costilla-Reyes, P. Scully, and K. B. Ozanyan. Deep neural networks for learning spatio-temporal features from tomography sensors. *IEEE Transactions on Industrial Electronics*, 65(1):645–653, 2018.

[21] S. Dabiri and K. Heaslip. Inferring transportation modes from gps trajectories using a convolutional neural network. *Transportation research part C: emerging technologies*, 86:360–371, 2018.

[22] Z. Diao, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He. Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting. In *AAAI*, 2019.

[23] D. Ding, M. Zhang, X. Pan, D. Wu, and P. Pu. Geographical feature extraction for entities in location-based social networks. In *WWW*, 2018.

[24] B. Du, H. Peng, S. Wang, M. Bhuiyan, L. Wang, Q. Gong, L. Liu, and J. Li. Deep irregular convolutional residual lstm for urban

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2020.3025580, IEEE Transactions on Knowledge and Data Engineering

traffic passenger flows prediction. *IEEE Transactions on Intelligent Transportation Systems*, 99:1–14, 2019.

[25] M. Du, N. Liu, and X. Hu. Techniques for interpretable machine learning. *arXiv:1808.00033 [cs.LG]*, 2018.

[26] L. Duan, T. Hu, E. Cheng, J. Zhu, and C. Gao. Deep convolutional neural networks for spatiotemporal crime prediction. In *IKE*, 2017.

[27] Y. Duan, Y. Lv, Y.-L. Liu, and F.-Y. Wang. An efficient realization of deep learning for traffic data imputation. *Transportation research part C: emerging technologies*, 72:168–181, 2016.

[28] N. C. Dvornek, P. Ventola, K. A. Pelphrey, and J. S. Duncan. Identifying autism from resting-state fmri using long short-term memory networks. In *International Workshop on Machine Learning in Medical Imaging*, 2017.

[29] Y. Endo, K. Nishida, H. Toda, and H. Sawada. Predicting destinations from partial trajectories using recurrent neural network. In *PAKDD*, 2017.

[30] Y. Endo, H. Toda, K. Nishida, and J. Ikedo. Classifying spatial trajectories using representation learning. *International Journal of Data Science and Analytics*, 2(3-4):107–117, 2016.

[31] Z. Fan, X. Song, T. Xia, R. Jiang, R. Shibasaki, and R. Sakuramachi. Online deep ensemble learning for predicting citywide human mobility. In *ACM IMWUT*, 2018.

[32] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *WWW*, 2018.

[33] T. Fernando, S. Denman, S. Sridharan, and C. Fookes. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural networks*, 108:466–478, 2018.

[34] T. G. C. N. for Traffic Speed Prediction Considering External Factors. Liang ge and hang li and junling liu and aoli zhou. In *MDM*, 2019.

[35] J. Gao, C. Xiao, Y. Wang, W. Tang, L. M. Glass, and J. Sun. Stagenet: Stage-aware neural networks for health risk prediction. In *The Web Conference*, 2020.

[36] Q. Gao, G. Trajcevski, F. Zhou, K. Zhang, T. Zhong, and F. Zhang. Trajectory-based social circle inference. In *SIGSPATIAL*, 2018.

[37] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, and F. Zhang. Identifying human mobility via trajectory embeddings. In *AAAI*, 2017.

[38] Y. Gao, L. Zhao, L. Wu, Y. Ye, H. Xiong, and C. Yang. Incomplete label multi-task deep learning for spatio-temporal event subtype forecasting. In *AAAI*, 2019.

[39] X. Geng, Y. Li, L. Wang, L. Zhang, J. Ye, Y. Liu, and Q. Yang. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In *AAAI*, 2019.

[40] S. Giffard-Roisin, M. Yang, G. Charpiat, B. Kégl, and C. Monteleoni. Fused deep learning for hurricane forecast fused deep learning for hurricane track forecast from reanalysis data. In *Climate Informatics Workshop*, 2018.

[41] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *AAAI*, 2019.

[42] X. Guo, K. C. Dominick, A. A. Minai, H. Li, C. A. Erickson, and L. J. Lu. Diagnosing autism spectrum disorder from brain resting-state functional connectivity patterns using a deep neural network with a novel feature selection method. *Frontiers in neuroscience*, 11:460, 2017.

[43] S. He and K. G. Shin. Spatio-temporal capsule-based reinforcement learning for mobility-on-demand network coordination. In *WWW*, 2019.

[44] Z. He, C.-Y. Chow, and J.-D. Zhang. Stcnn: A spatio-temporal convolutional neural network for long-term traffic prediction. In *MDM*, 2019.

[45] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz, and F. Meneguzzi. Identification of autism spectrum disorder using deep learning and the abide dataset. *NeuroImage: Clinical*, 17:16–23, 2018.

[46] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2013.

[47] T. Horikawa and Y. Kamitani. Generic decoding of seen and imagined objects using hierarchical visual features. *Nature communications*, 8:15037, 2017.

[48] M. Hossain, B. Rekabdar, S. J. Louis, and S. Dascalu. Forecasting the weather of nevada: A deep learning approach. In *IJCNN*, 2015.

[49] C. Huang, C. Zhang, J. Zhao, X. Wu, N. Chawla, and D. Yin. Mist: A multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting. In *WWW*, 2019.

[50] C. Huang, J. Zhang, Y. Zheng, and N. V. Chawla. Deepcrime: Attentive hierarchical recurrent networks for crime prediction. In *CIKM*, 2018.

[51] H. Huang, X. Hu, J. Han, J. Lv, N. Liu, L. Guo, and T. Liu. Latent source mining in fmri data via deep neural network. In *ISBI*, 2016.

[52] H. Huang, X. Hu, Y. Zhao, M. Makkie, Q. Dong, S. Zhao, L. Guo, and T. Liu. Modeling task fmri data via deep convolutional autoencoder. *IEEE transactions on medical imaging*, 37(7):1551–1561, 2018.

[53] W. Huang, G. Song, H. Hong, and K. Xie. Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. *IEEE Trans. Intelligent Transportation Systems*, 15(5):2191–2201, 2014.

[54] R. J. Huster, S. Debener, T. Eichele, and C. S. Herrmann. Methods for simultaneous eeg-fmri: An introductory review. *Journal of Neuroscience*, 32(18):6053–6060, 2012.

[55] H. Jang, S. M. Plis, V. D. Calhoun, and J.-H. Lee. Task-specific feature extraction and classification of fmri volumes using a deep neural network initialized with a deep belief network: Evaluation using sensorimotor tasks. *NeuroImage*, 145:314–328, 2017.

[56] R. Jiang, X. Song, Z. Fan, T. Xia, Q. Chen, Q. Chen, and R. Shibasaki. Deep roi-based modeling for urban human mobility prediction. In *ACM IMWUT*, 2018.

[57] M. Jin, T. Curran, and D. Cordes. Classification of amnestic mild cognitive impairment using fmri. In *ISBI*, pages 29–32, 2014.

[58] A. Karatzoglou, N. Schnell, and M. Beigl. A convolutional neural network approach for modeling semantic trajectories and predicting future locations. In *ICANN*, pages 61–72, 2018.

[59] J. Kawahara, C. J. Brown, S. P. Miller, B. G. Booth, V. Chau, R. E. Grunau, J. G. Zwicker, and G. Hamarneh. Brainnetcnn: convolutional neural networks for brain networks; towards predicting neurodevelopment. *NeuroImage*, 146:1038–1049, 2017.

[60] J. Ke, H. Yang, H. Zheng, X. Chen, Y. Jia, P. Gong, and J. Ye. Hexagon-based convolutional neural network for supply-demand forecasting of ride-sourcing services. *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[61] J. Ke, H. Zheng, H. Yang, and X. M. Chen. Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation Research Part C: Emerging Technologies*, 85:591–608, 2017.

[62] J. Kim, V. D. Calhoun, E. Shim, and J.-H. Lee. Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance: Evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. *Neuroimage*, 124:127–146, 2016.

[63] S. Kim, S. Ames, J. Lee, C. Zhang, A. C. Wilson, and D. Williams. Resolution reconstruction of climate data with pixel recursive model. In *ICDMW*, 2017.

[64] S. Kim, S. Hong, M. Joh, and S.-k. Song. Deeprain: Convlstm network for precipitation prediction using multichannel radar data. *arXiv preprint arXiv:1711.02316*, 2017.

[65] Z. Kira, W. Li, R. Allen, and A. R. Wagner. Leveraging deep learning for spatio-temporal understanding of everyday environments. In *IJCAI Workshop on Deep Learning and Artificial Intelligence*, 2016.

[66] S. Kisilevich, F. Mansmann, M. Nanni, and S. Rinzivillo. Spatio-temporal clustering: a survey. *Data Mining and Knowledge Discovery Handbook, Springer*, 2015.

[67] J. Kleesiek, G. Urban, A. Hubert, D. Schwarz, K. Maier-Hein, M. Bendszus, and A. Biller. Deep mri brain extraction: a 3d convolutional neural network for skull stripping. *NeuroImage*, 129:460–469, 2016.

[68] D. Kong and F. Wu. Hst-lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In *IJCAI*, 2018.

[69] S. Korolev, A. Safiullin, M. Belyaev, and Y. Dodonova. Residual and plain convolutional neural networks for 3d brain mri classification. In *ISBI*, pages 835–838, 2017.

[70] T. Kurth, S. Treichler, J. Romero, M. Mudigonda, N. Luehr, E. Phillips, A. Mahesh, M. Matheson, J. Deslippe, M. Fatica, et al. Exascale deep learning for climate analytics. In *SC*, 2018.

[71] D. Lee, S. Jung, Y. Cheon, D. Kim, and S. You. Forecasting taxi demands with fully convolutional networks and temporal guided embedding. In *NIPS Workshop on Modeling and Decision-Making in the Spatiotemporal Domain*, 2018.

[72] R. Li, Y. Shen, and Y. Zhu. Next point-of-interest recommendation with temporal and multi-level context attention. In *ICDM*, 2018.

[73] X. Li, K. Zhao, G. Cong, C. S. Jensen, and W. Wei. Deep representation learning for trajectory similarity computation. In *ICDE*, 2018.

[74] Y. Li, K. Fu, Z. Wang, C. Shahabi, J. Ye, and Y. Liu. Multi-task representation learning for travel time estimation. In *KDD*, 2018.

[75] Y. Li and B. Shuai. Origin and destination forecasting on dockless shared bicycle in a hybrid deep-learning algorithms. *Multimedia Tools and Applications*, pages 1–12, 2018.

[76] Y. Li, R. Yu, C. Shahabi, and Y. Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *ICLR*, 2018.

[77] Y. Li, Z. Zhu, D. Kong, M. Xu, and Y. Zhao. Learning heterogeneous spatial-temporal representation for bike-sharing demand prediction. In *AAAI*, 2019.

[78] Z. Li. Spatiotemporal pattern mining: Algorithms and applications. *Frequent Pattern Mining*, 2014.

[79] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng. Geoman: Multi-level attention networks for geo-sensory time series prediction. In *IJCAI*, pages 3428–3434, 2018.

[80] B. Liao, J. Zhang, M. Cai, S. Tang, Y. Gao, C. Wu, S. Yang, W. Zhu, Y. Guo, and F. Wu. Dest-resnet: A deep spatiotemporal residual network for hotspot traffic speed prediction. In *ACM MM*, 2018.

[81] D. Liao, W. Liu, Y. Zhong, J. Li, and G. Wang. Predicting activity and location with multi-task context aware recurrent neural network. In *IJCAI*, pages 3435–3441, 2018.

[82] L. Lin, Z. He, and S. Peeta. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research Part C: Emerging Technologies*, 97:258–276, 2018.

[83] Y. Lin, X. Dai, L. Li, and F.-Y. Wang. Pattern sensitive prediction of traffic flow based on generative adversarial framework. *IEEE Transactions on Intelligent Transportation Systems*, (99):1–6, 2018.

[84] Y. Lin, N. Mago, Y. Gao, Y. Li, Y.-Y. Chiang, C. Shahabi, and J. L. Ambite. Exploiting spatiotemporal patterns for accurate air quality forecasting using deep learning. In *SIGSPATIAL*, 2018.

[85] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin. Deepstn+: Contextaware spatial temporal neural network for crowd flow prediction in metropolis. In *AAAI*, 2019.

[86] H. Liu, X. Mi, and Y. Li. Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network. *Energy Conversion and Management*, 166:120–131, 2018.

[87] H. Liu, X. Mi, and Y. Li. Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, lstm network and elm. *Energy Conversion and Management*, 159:54–64, 2018.

[88] L. Liu, Z. Qiu, G. Li, Q. Wang, W. Ouyang, and L. Lin. Contextualized spatial–temporal network for taxi origin-destination demand prediction. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3875 – 3887, 2019.

[89] L. Liu, R. Zhang, J. Peng, G. Li, B. Du, and L. Lin. Attentive crowd flow machines. *arXiv preprint arXiv:1809.00101*, 2018.

[90] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*, 2016.

[91] Y. Liu, E. Racah, J. Correa, A. Khosrowshahi, D. Lavers, K. Kunkel, M. Wehner, W. Collins, et al. Application of deep convolutional neural networks for detecting extreme weather in climate datasets. *arXiv preprint arXiv:1605.01156*, 2016.

[92] Y. Liu, Y. Wang, X. Yang, and L. Zhang. Short-term travel time prediction by deep learning: A comparison of different lstm-dnn models. In *ITSC*, 2017.

[93] Z. Liu, Z. Li, K. Wu, and M. Li. Urban traffic prediction from mobility data using deep learning. *IEEE Network*, 32(4):40–46, 2018.

[94] J. Lv, Q. Li, Q. Sun, and X. Wang. T-conv: A convolutional neural network for multi-scale taxi trajectory prediction. In *BigComp*, 2018.

[95] Y. Lv, Y. Duan, W. Kang, Z. Li, F.-Y. Wang, et al. Traffic flow prediction with big data: A deep learning approach. *IEEE Trans. Intelligent Transportation Systems*, 16(2):865–873, 2015.

[96] Z. Lv, J. Xu, K. Zheng, H. Yin, P. Zhao, and X. Zhou. Lc-rnn: A deep learning model for traffic speed prediction. In *IJCAI*, 2018.

[97] L. Ma, J. Gao, Y. Wang, C. Zhang, J. Wang, W. Ruan, W. Tang, X. Gao, and X. Ma. Adacare: Explainable clinical health status representation learning via scale-adaptive feature extraction and recalibration. In *AAAI*, 2020.

[98] L. Ma, C. Zhang, Y. Wang, W. Ruan, J. Wang, W. Tang, X. Ma, X. Gao, and J. Gao. Concare: Personalized clinical feature embedding via capturing the healthcare context. In *AAAI*, 2020.

[99] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4):818, 2017.

[100] X. Ma, H. Yu, Y. Wang, and Y. Wang. Large-scale transportation network congestion evolution prediction using deep learning theory. *PloS one*, 10(3):e0119044, 2015.

[101] X. Ma, J. Zhang, B. Du, C. Ding, and L. Sun. Parallel architecture of convolutional bi-directional lstm neural networks for network-wide metro ridership prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[102] A. H. Marblestone, G. Wayne, and K. P. Kording. Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10:94, 2016.

[103] H. Martin, D. Bucher, E. Suel, P. Zhao, F. Perez-Cruz, and M. Raubal. Graph convolutional neural networks for human activity purpose imputation from gps-based trajectory data. In *NIPS Workshop on Modeling and Decision-Making in the Spatiotemporal Domain*, 2018.

[104] J. D. Mazimpaka and S. Timpf. Trajectory data mining: A review of methods and applications. *Journal of Spatial Information Science*, (13):61–99, 2016.

[105] R. J. Meszlényi, K. Buza, and Z. Vidnyánszky. Resting state fmri functional connectivity-based classification using a convolutional neural network architecture. *Frontiers in neuroinformatics*, 11:61, 2017.

[106] H. Nguyen, L.-M. Kieu, T. Wen, and C. Cai. Deep learning methods in transportation domain: a review. *IET Intelligent Transport Systems*, 12(9):998–1004, 2018.

[107] N. T. Nguyen, Y. Wang, H. Li, X. Liu, and Z. Han. Extracting typical users' moving patterns using deep learning. In *GLOBECOM*, 2012.

[108] D. Nie, H. Zhang, E. Adeli, L. Liu, and D. Shen. 3d deep learning for multi-modal imaging-guided survival time prediction of brain tumor patients. In *MICCAI*, 2016.

[109] X. Niu, Y. Zhu, and X. Zhang. Deepsense: A novel learning mechanism for traffic prediction with taxi gps traces. In *GLOBECOM*, 2014.

[110] X. Ouyang, C. Zhang, P. Zhou, and H. Jiang. Deepspace: An online deep learning framework for mobile big data to understand human mobility patterns. *arXiv preprint arXiv:1610.07009*, 2016.

[111] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang. Urban tra ic prediction from spatio-temporal data using deep meta learning. In *KDD*, 2019.

[112] J. Pang, J. Huang, Y. Du, H. Yu, Q. Huang, and B. Yin. Learning to predict bus arrival time from heterogeneous measurements via recurrent neural network. *IEEE Transactions on Intelligent Transportation Systems*, 20(9):3283 – 3293, 2019.

[113] P. Patel, P. Aggarwal, and A. Gupta. Classification of schizophrenia versus normal subjects using deep learning. In *ICVGIP*, 2016.

[114] S. M. Plis, D. R. Hjelm, R. Salakhutdinov, E. A. Allen, H. J. Bockholt, J. D. Long, H. J. Johnson, J. S. Paulsen, J. A. Turner, and V. D. Calhoun. Deep learning for neuroimaging: a validation study. *Frontiers in neuroscience*, 8:229, 2014.

[115] N. G. Polson and V. O. Sokolov. Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, 79:1–17, 2017.

[116] E. Racah, C. Beckham, T. Maharaj, S. E. Kahou, M. Prabhat, and C. Pal. Extremeweather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events. In *NIPS*, pages 3402–3413, 2017.

[117] S. Rasp and S. Lerch. Neural networks for post-processing ensemble weather forecasts. *arXiv preprint arXiv:1805.09091*, 2018.

[118] H. Ren, Y. Song, J. Wang, Y. Hu, and J. Lei. A deep learning approach to the citywide traffic accident risk prediction. In *ITSC*, 2018.

[119] F. Rodrigues, I. Markou, and F. C. Pereira. Combining time-series and textual data for taxi demand prediction in event areas: a deep learning approach. *Information Fusion*, 49:120–129, 2019.

[120] I. Roesch and T. Günther. Visualization of neural network predictions for weather forecasting. In *Computer Graphics Forum*. Wiley Online Library, 2017.

[121] S. Sarraf and G. Tofighi. Deep learning-based pipeline to recognize alzheimer's disease using fmri data. In *Future Technologies Conference*, pages 816–820. IEEE, 2016.

[122] S. Scher. Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning. *Geophysical Research Letters*, 45(22):12–616, 2018.

[123] S. Shekhar, Z. Jiang, R. Y. Ali, E. Eftelioglu, X. Tang, V. M. V. Gunturi, and X. Zhou. Spatiotemporal data mining: A computational perspective. *ISPRS International Journal of Geo-Information*, 4:2306–2338, 2015.

[124] B. Shen, X. Liang, Y. Ouyang, M. Liu, W. Zheng, and K. M. Carley. Stepdeep: A novel spatial-temporal mobility event prediction framework based on deep neural network. In *KDD*, 2018.

[125] J. Shi, X. Zheng, Y. Li, Q. Zhang, and S. Ying. Multimodal neuroimaging feature learning with multimodal stacked deep polynomial networks for diagnosis of alzheimer's disease. *IEEE journal of biomedical and health informatics*, 22(1):173–183, 2018.

[126] X.-J. Shi, Z.-H. Gao, L. Lausen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. Deep learning for precipitation nowcasting: A benchmark and a new model. In *NIPS*, pages 5617–5627, 2017.

[127] Y. Shi, Y. Tian, Y. Wang, and T. Huang. Sequential deep trajectory descriptor for action recognition with three-stream cnn. *IEEE Transactions on Multimedia*, 19(7):1510–1520, 2017.

[128] X. Song, H. Kanasugi, and R. Shibasaki. Deeptransport: Prediction and simulation of human mobility and transportation mode at a citywide level. In *IJCAI*, 2016.

[129] R. Soua, A. Koesdwiady, and F. Karray. Big-data-generated traffic flow prediction using deep learning and dempster-shafer theory. In *IJCNN*, pages 3195–3202. IEEE, 2016.

[130] F. Sun, A. Dubey, and J. White. Dxnat-deep neural networks for explaining non-recurring traffic congestion. In *IEEE International Conference on Big Data*, 2017.

[131] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.

[132] Y. Tao, X. Gao, A. Ihler, K. Hsu, and S. Sorooshian. Deep neural networks for precipitation estimation from remotely sensed information. In *CEC*, 2016.

[133] R. R. Vatsavai, V. Chandola, S. Klasky, A. Ganguly, A. Stefanidis, and S. Shekhar. Spatiotemporal data mining in the era of big spatial data: algorithms and applications. In *SIGSPATIAL international workshop on analytics for big geospatial data*, 2012.

[134] B. Wang, X. Luo, F. Zhang, B. Yuan, A. L. Bertozzi, and P. J. Brantingham. Graph-based deep modeling and real time forecasting of sparse spatio-temporal data. *preprint arXiv:1804.00684*, 2018.

[135] B. Wang, P. Yin, A. L. Bertozzi, P. J. Brantingham, S. J. Osher, and J. Xin. Deep learning for real-time crime forecasting and its ternarization. *arXiv preprint arXiv:1711.08833*, 2017.

[136] D. Wang, W. Cao, J. Li, and J. Ye. Deepsd: supply-demand prediction for online car-hailing services using deep neural networks. In *ICDE*, pages 243–254. IEEE, 2017.

[137] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng. When will you arrive? estimating travel time based on deep neural networks. AAAI, 2018.

[138] H. Wang, G. Liu, J. Duan, and L. Zhang. Detecting transportation modes using deep neural network. *IEICE TRANSACTIONS on Information and Systems*, 100(5):1132–1135, 2017.

[139] J. Wang, Q. Gu, J. Wu, G. Liu, and Z. Xiong. Traffic speed prediction and congestion source exploration: A deep learning method. In *ICDM*, 2016.

[140] J. Wang, X. He, Z. Wang, J. Wu, N. J. Yuan, X. Xie, and Z. Xiong. Cd-cnn: A partially supervised cross-domain deep learning model for urban resident recognition. In *AAAI*, 2018.

[141] J. Wang, N. Wu, X. Lu, X. Zhao, and K. Feng. Deep trajectory recovery with fine-grained calibration using kalman filter. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[142] J. Wang, N. Wu, W. X. Zhao, F. Peng, and X. Lin. Empowering a* search algorithms with neural networks for personalized route recommendation. In *KDD*, 2019.

[143] L. Wang, X. Geng, X. Ma, F. Liu, and Q. Yang. Cross-city transfer learning for deep spatio-temporal prediction. In *IJCAI*, 2018.

[144] P. Wang, Y. Fu, J. Zhang, X. Li, and D. Lin. Learning urban community structures: A collective embedding perspective with periodic spatial-temporal mobility graphs. *ACM Transactions on Intelligent Systems and Technology*, 9(6):63, 2018.

[145] P. Wang, Z. Li, Y. Hou, and W. Li. Action recognition based on joint trajectory maps using convolutional neural networks. In *ACM MM*, 2016.

[146] S. Wang, X. Zhang, F. Li, P. S. Yu, and Z. Huang. Efficient traffic estimation with multi-sourced data by parallel coupled hidden markov model. *IEEE Trans. Intell. Transp. Syst.*, 20(8), 2019.

[147] X. Wang, C. Chen, Y. Min, J. He, B. Yang, and Y. Zhang. Efficient metropolitan traffic prediction based on graph recurrent neural network. *arXiv preprint arXiv:1811.00740*, 2018.

[148] Y. Wang, M. Long, J. Wang, Z. Gao, and S. Y. Philip. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *NIPS*, pages 879–888, 2017.

[149] Y. Wang, H. Yin, H. Chen, T. Wo, J. Xu, and K. Zheng. Origin-destination matrix prediction via graph convolution: a new perspective of passenger demand modeling. In *KDD*, 2019.

[150] Y. Wang, D. Zhang, Y. Liu, B. Dai, and L. H. Lee. Enhancing transportation systems via deep learning: A survey. *Transportation Research Part C: Emerging Technologies*, 2018.

[151] D. Wen, Z. Wei, Y. Zhou, G. Li, X. Zhang, and W. Han. Deep learning methods to process fmri data and their application in the diagnosis of cognitive impairment: A brief overview and our opinion. *Frontiers in neuroinformatics*, 12:23, 2018.

[152] H. Wu, Z. Chen, W. Sun, B. Zheng, and W. Wang. Modeling trajectories with recurrent neural networks. In *IJCAI*, 2017.

[153] Z. Wu, S. Pan, F. Chen, G. Long, and P. S. Y. Chengqi Zhang. A comprehensive survey on graph neural networks. *arXiv: 1901.00596v2*, 2019.

[154] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NIPS*, 2015.

[155] C. Xu, J. Ji, and P. Liu. The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets. *Transportation Research Part C: Emerging Technologies*, 95:47–60, 2018.

[156] K. Xu, Z. Qin, G. Wang, K. Huang, S. Ye, and H. Zhang. Collision-free lstm for human trajectory prediction. In *MMM*, 2018.

[157] S. Xu, J. Xu, R. Zhou, C. Liu, Z. Li, and A. Liu. Tadnm: A transportation-mode aware deep neural model for travel time estimation. In *DASFAA*, 2020.

[158] C. Yang, M. Sun, W. X. Zhao, Z. Liu, and E. Y. Chang. A neural network approach to jointly modeling social networks and mobile trajectories. *ACM Transactions on Information Systems*, 35(4):36, 2017.

[159] G. Yang, Y. Cai, and C. K. Reddy. Recurrent spatio-temporal point process for check-in time prediction. In *CIKM*, 2018.

[160] G. Yang, Y. Cai, and C. K. Reddy. Spatio-temporal check-in time prediction with recurrent neural network based survival analysis. In *IJCAI*, 2018.

[161] H.-F. Yang, T. S. Dillon, and Y.-P. P. Chen. Optimized structure of the traffic flow forecasting model with a deep learning approach. *IEEE transactions on neural networks and learning systems*, 28(10):2371–2381, 2017.

[162] J. Yang and C. Eickhoff. Unsupervised learning of parsimonious general-purpose embeddings for user and location modeling. *ACM Transactions on Information Systems*, 36(3):32, 2018.

[163] D. Yao, C. Zhang, J. Huang, and J. Bi. Serm: A recurrent model for next location prediction in semantic trajectories. In *CIKM*, 2017.

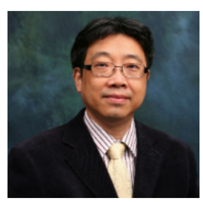
[164] D. Yao, C. Zhang, Z. Zhu, Q. Hu, Z. Wang, J. Huang, and J. Bi. Learning deep representation for trajectory clustering. *Expert Systems*, 35(2):e12252, 2018.

[165] D. Yao, C. Zhang, Z. Zhu, J. Huang, and J. Bi. Trajectory clustering via deep representation learning. In *IJCNN*, 2017.

[166] H. Yao, Y. Liu, Y. Wei, X. Tang, and Z. Li. Learning from multiple cities: A meta-learning approach for spatial-temporal prediction. In *WWW*, 2019.

[167] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *AAAI*, 2019.
[168] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, and J. Ye. Deep multi-view spatial-temporal network for taxi demand prediction. In *AAAI*, 2018.
[169] J. Ye, L. Sun, B. Du, Y. Fu, X. Tong, and H. Xiong. Co-prediction of multiple transportation demands based on deep spatio-temporal neural network. In *KDD*, 2019.
[170] H. Yin, Z. Hu, X. Zhou, H. Wang, K. Zheng, Q. V. H. Nguyen, and S. Sadiq. Discovering interpretable geo-social communities. In *ICDE*, 2016.
[171] H. Yin, W. Wang, H. Wang, L. Chen, and X. Zhou. Spatial-aware hierarchical collaborative deep learning for poi recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 29(11):2537 – 2551, 2017.
[172] B. Yu, H. Yin, and Z. Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *IJCAI*, 2018.
[173] R. Yu, Y. Li, C. Shahabi, U. Demiryurek, and Y. Liu. Deep learning: A generic approach for extreme condition traffic forecasting. In *SDM*, 2017.
[174] R. Yuecheng, X. Zhimian, Y. Ruibo, and M. Xu. Du-parking: Spatio-temporal big data tells you realtime parking availability. In *KDD*, 2018.
[175] M. A. Zaytar and C. El Amrani. Sequence to sequence weather forecasting with long short term memory recurrent neural networks. *Int J Comput Appl*, 143(11), 2016.
[176] G. Zhan. A semantic sequential correlation based lstm model for next poi recommendation. In *MDM*, 2019.
[177] H. Zhang, H. Wu, W. Sun, and B. Zheng. Deeptravel: a neural network based travel time estimation model with auxiliary supervision. In *IJCAI*, 2018.
[178] J. Zhang, B. Cao, S. Xie, C.-T. Lu, P. S. Yu, and A. B. Ragin. Identifying connectivity patterns for brain diseases via multi-side-view guided deep architectures. In *SDM*, 2016.
[179] J. Zhang, Y. Zheng, and D. Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *AAAI*, 2017.
[180] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi. Dnn-based prediction model for spatio-temporal data. In *SIGSPATIAL*, 2016.
[181] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, and T. Li. Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artificial Intelligence*, 259:147–166, 2018.
[182] P. Zhang, Y. Jia, J. Gao, W. Song, and H. K. Leung. Short-term rainfall forecasting using multi-layer perceptron. *IEEE Transactions on Big Data*, 2018.
[183] W. Zhang, L. Han, J. Sun, H. Guo, and J. Dai. Application of multi-channel 3d-cube successive convolution network for convective storm nowcasting. *preprint arXiv:1702.04517*, 2017.
[184] Y. Zhang, S. Wang, B. Chen, and J. Cao:. Gcgan: Generative adversarial nets with graph cnn for network-scale traffic prediction. In *IJCNN*, 2019.
[185] Y. Zhang, S. Wang, B. Chen, J. Cao, and Z. Huang. Trafficgan: Network-scale deep traffic prediction with generative adversarial nets. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–12, December 2019.
[186] Z. Zhang, Q. He, J. Gao, and M. Ni. A deep learning approach for detecting traffic accidents from social media data. *Transportation research part C: emerging technologies*, 86:580–596, 2018.
[187] J. Zhao, J. Xu, R. Zhou, P. Zhao, C. Liu, and F. Zhu. On prediction of user destination by sub-trajectory understanding: A deep learning based approach. In *CIKM*, 2018.
[188] P. Zhao, H. Zhu, Y. Liu, J. Xu, Z. Li, F. Zhuang, and V. S. Sheng. Where to go next: A spatio-temporal gated network for next poi recommendation. In *AAAI*, 2019.
[189] S. Zhao, T. Zhao, I. King, and M. R. Lyu. Geo-teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation. In *WWW*, pages 153–162, 2017.
[190] Y. Zhao, Q. Dong, S. Zhang, W. Zhang, H. Chen, X. Jiang, L. Guo, X. Hu, J. Han, and T. Liu. Automatic recognition of fmri-derived functional networks using 3-d convolutional neural networks. *IEEE Transactions on Biomedical Engineering*, 65(9):1975–1984, 2018.
[191] S. Zheng, Y. Yue, and J. Hobbs. Generating long-term trajectories using deep hierarchical networks. In *NIPS*, 2016.
[192] Y. Zheng. Tutorial on location-based social networks. In *WWW*, 2012.
[193] F. Zhou, Q. Gao, G. Trajcevski, K. Zhang, T. Zhong, and F. Zhang. Trajectory-user linking via variational autoencoder. In *IJCAI*, 2018.
[194] N. Zhou, W. X. Zhao, X. Zhang, J. Wen, and S. Wang. A general multi-context embedding model for mining human trajectory data. *IEEE Transactions on Knowledge and Data Engineering*, 28(8):1945–1958, 2016.
[195] X. Zhou, Y. Shen, Y. Zhu, and L. Huang. Predicting multi-step citywide passenger demands using attention-based neural networks. In *WSDM*, 2018.
[196] L. Zhu, F. Guo, R. Krishnan, and J. W. Polak. A deep learning approach for traffic incident detection in urban networks. In *ITSC*, 2018.
[197] Q. Zhu, J. Chen, L. Zhu, X. Duan, and Y. Liu. Wind speed prediction with spatio–temporal correlation: A deep learning approach. *Energies*, 11(4):705, 2018.
[198] Y. Zhuoning, Z. Xun, and Y. Tianbao. Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *KDD*, 2018.
[199] A. Zonoozi, J.-j. Kim, X.-L. Li, and G. Cong. Periodic-crn: A convolutional recurrent model for crowd density prediction with recurring periodic patterns. In *IJCAI*, 2018.

Senzhang Wang received the B.Sc. and Ph.D. degree in Southeast University, Nanjing, China in 2009 and Beihang University, Beijing, China in 2016 respectively. He is currently an associate professor at College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. His main research focus is on data mining, social computing, and urban computing. He has published more than 80 referred conference and journal papers.



Jiannong Cao received the B.Sc. degree in computer science from Nanjing University, China, in 1982, and the M.Sc. and Ph.D. degrees in computer science from Washington State University, USA, in 1986 and 1990 respectively. He is currently a Chair Professor of Department of Computing at The Hong Kong Polytechnic University, Hong Kong. His research interests include parallel and distributed computing, wireless networks and mobile computing, big data and cloud computing, pervasive computing, and fault tolerant computing. He has published over 500 papers in major international journals and conference proceedings. He is a fellow of IEEE, a distinguished member of ACM, a senior member of China Computer Federation (CCF).

Philip S. Yu received the B.S. Degree in E.E. from National Taiwan University, the M.S. and Ph.D. degrees in E.E. from Stanford University, and the M.B.A. degree from New York University. He is a Distinguished Professor in Computer Science at the University of Illinois at Chicago and also holds the Wexler Chair in Information Technology. Before joining UIC, Dr. Yu was with IBM, where he was manager of the Software Tools and Techniques department at the Watson Research Center. His research interest is on big data, including data mining, data stream, database and privacy. He has published more than 1,100 papers in refereed journals and conferences. He holds or has applied for more than 300 US patents. Dr. Yu is a Fellow of the ACM and the IEEE.